PassQueen
a.k.a dumpsqueen.com

NetApp Certified Hybrid Cloud Implementation Engineer

Netapp NS0-402

Version Demo

Total Demo Questions: 8

Total Premium Questions: 82

Buy Premium PDF

https://passqueen.com

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Cloud Fundamentals	29
Topic 2, Cloud Volumes ONTAP	36
Topic 3, StorageGRID	2
Topic 4, Cloud Manager	10
Topic 5, Mix Questions	5
Total	82

QUESTION NO: 1

You must protect an application SVM and its associated data volumes at a disaster recovery site. The recovery SVM must retain the source SVM configuration when it is activated.

In this scenario, which ONTAP replication technology should you use?

A. SnapVault

B. SVM disaster recovery

C. FlexCache

D. MetroCluster IP

ANSWER: B

Explanation:

SVM disaster recovery is correct because it replicates an entire Storage Virtual Machine, including its data volumes and supported SVM configuration, to a destination cluster. This provides a recoverable SVM-level copy that can be activated at the disaster recovery site when the source environment is unavailable.

SVM disaster recovery uses SnapMirror technology but protects the SVM as the replication unit rather than requiring each volume relationship to be configured and recovered independently. This makes it appropriate when both the data and the SVM configuration are required for recovery. See the ONTAP SVM disaster recovery documentation.

QUESTION NO: 2

You are writing a script to verify, update and display discrepancies of IT assets in the corporate content management database (CMDB). You must use the RESTful API to interface with the CMDB server.

What are two common formats for the RESTful API responses? (Choose two.)

A. MarkDown

B. JSON

C. XML

D. SGML

ANSWER: B C

Explanation:

JSON and XML are common representation formats for RESTful API request and response bodies. JSON is especially prevalent in modern REST APIs because its lightweight object-and-array structure maps naturally to application data and is broadly supported by scripting languages. API clients typically identify JSON through the `application/json` media type in the HTTP `Content-Type` or `Accept` headers. XML is also a well-established API response format, particularly in enterprise integrations and older or standards-based web services. XML responses are generally identified by media types such as `application/xml` or `text/xml`. REST is an architectural style that uses HTTP semantics and representations of resources; it does not require one single serialization format. A CMDB REST API can therefore expose asset records, configuration items, relationships, and discrepancy results as either JSON or XML when the server supports content negotiation or explicitly defines those formats. The HTTP semantics specification describes representation metadata and content negotiation, while the JSON RFC defines the JSON media type and interchange format. See RFC 9110: HTTP Semantics and RFC 8259: The JSON Data Interchange Format.

QUESTION NO: 3

Your company asks you to deploy a new version control management system. In this scenario, which two tools would accomplish this task? (Choose two.)

A. Confluence

B. GitHub

C. ALM

D. SourceForge

ANSWER: B D

Explanation:

GitHub and SourceForge are appropriate tools for deploying a version control management capability. GitHub provides hosted Git repositories, enabling teams to store source code, track changes over time, manage branches, merge contributions through pull requests, and apply repository access controls. These capabilities make it a commonly used platform for collaborative source-code version control in development and DevOps environments.

SourceForge also provides project hosting with source-code repositories and version-control support. It can be used to host software projects, maintain revision history, and enable developers to collaborate on managed codebases. As a result, it meets the core requirement of a version control management system: preserving versions of files and allowing controlled, traceable team changes.

In a NetApp solution context, version-control platforms are typically part of the development toolchain whose repository data can be stored, protected, replicated, and recovered using NetApp storage and data-management capabilities. The essential requirement in this scenario is the repository and revision-management function supplied by GitHub and SourceForge. See GitHub Features and SourceForge.

QUESTION NO: 4

Which three Trident storage drivers support clone operations? (Choose three.)

A. aws-cvs

B. ontap-san

C. ontap-nas-economy

D. ontap-nas-flexgroup

E. ontap-nas

ANSWER: B C E

Explanation:

Trident clone operations are supported by the ONTAP-backed `ontap-san`, `ontap-nas-economy`, and `ontap-nas` drivers. For these drivers, Trident can provision a new persistent volume from an existing source PVC by using ONTAP cloning capabilities, allowing the new volume to begin as a point-in-time copy of the source data. This is useful for development and test workflows, data refreshes, and application deployment patterns that require a writable copy of an existing dataset without first performing a full host-side copy.

The source and destination are handled through Kubernetes PVC cloning semantics, while Trident translates the request into the appropriate ONTAP storage operation. The clone is created within the storage constraints applicable to the driver and backend configuration, including compatible storage class and ONTAP placement requirements. The `ontap-san` driver provides this capability for SAN-provisioned volumes, while `ontap-nas` and `ontap-nas-economy` provide it for NAS-provisioned volumes. NetApp's Trident documentation lists the supported drivers and describes PVC cloning behavior in its volume-object and Kubernetes cloning documentation: Trident 18.10 volume objects documentation and Astra Trident volume cloning documentation.

QUESTION NO: 5

You are asked to design a Trident solution on NetApp ONTAP for a highly scaled Kubernetes environment that contains thousands of stateful microservice applications.

Which ONTAP Trident driver should be used in such a highly scaled environment?

A. ontap-nas-flexgroup

B. ontap-nas

C. ontap-nas-economy

D. ontap-san

ANSWER: C

Explanation:

`ontap-nas-economy` is the appropriate driver for a Kubernetes environment requiring very high persistent-volume density. This driver is specifically designed to provision NAS storage economically at scale by creating Trident volumes as qtrees within shared ONTAP FlexVol volumes, rather than requiring a separate FlexVol volume for every persistent volume claim. That architecture substantially reduces the number of FlexVol volumes that must be managed while still providing each Kubernetes application with an independently provisioned NFS-backed persistent volume. For an environment containing thousands of stateful microservices, this density model helps keep ONTAP object counts and operational overhead practical, while allowing Trident to dynamically provision storage through Kubernetes StorageClasses. Capacity planning must account for the shared FlexVol containers and qtree limits, but the economy driver is the intended selection when the primary design requirement is a very large number of NAS persistent volumes. NetApp's driver documentation describes `ontap-nas-economy` as the ONTAP NAS driver that uses qtrees and is intended for environments requiring higher volume density. See the NetApp Trident ONTAP NAS driver documentation and the ONTAP NFS administration documentation.

QUESTION NO: 6

You are automating the build process of several NetApp ONTAP Select instances. You want to automate the installation of license keys using the ONTAP REST API that is expecting an array as valid input.

Which automation process is correct in this scenario?

A.
```
{
  "keys":
["KQSRRRRRYVHXCFABGAAAAAAAAAAA, SOHOURRRYVHXCFABGAAAAAAAAAAA
EJFDVRRRYVHXCFABGAAAAAAAAAAA"]
}
```

B.
```
{
  keys:
"KQSRRRRRYVHXCFABGAAAAAAAAAAA, SOHOURRRYVHXCFABGAAAAAAAAAAA,
EJFDVRRRYVHXCFABGAAAAAAAAAAA"
}
```

C.
```
{
  "keys": @ ("KQSRRRRRYVHXCFABGAAAAAAAAAAA,
SOHOURRRYVHXCFABGAAAAAAAAAAA, EJFDVRRRYVHXCFABGAAAAAAAAAAA")
}
```

D.
```
{
  "key1": KQSRRRRRYVHXCFABGAAAAAAAAAAA
  "key2": SOHOURRRYVHXCFABGAAAAAAAAAAA
  "key3": EJFDVRRRYVHXCFABGAAAAAAAAAAA
}
```

A. Option A

B. Option B

C. Option C

D. Option D

ANSWER: B

Explanation:

Option B is correct because the automation must submit the license-key values as one JSON array in the REST request body. When an ONTAP REST API resource defines a property as an array, the client must preserve that structure during serialization: each license key is an individual array element, enclosed by square brackets, rather than being submitted as separate scalar requests or as one concatenated string. This is especially important in automated deployments, where a variable containing multiple keys must be rendered into valid JSON without changing its data type. The process represented by Option B maintains the required collection structure and therefore allows the API to process the complete set of license keys in a single valid payload. Automation tooling such as Ansible must similarly use JSON serialization or an equivalent native list representation so the HTTP request has an `application/json` body with an array-valued license-key field. NetApp's ONTAP REST API documentation describes REST resources and their JSON request-body conventions, while the ONTAP Select documentation provides deployment and licensing context. See NetApp ONTAP REST API documentation and NetApp ONTAP Select documentation.

QUESTION NO: 7

You are planning FabricPool capacity tiering for an on-premises ONTAP cluster. The cloud tier must use a supported public-cloud object storage service.

In this scenario, which two object storage services can be used as a FabricPool cloud tier? (Choose two.)

A. Amazon Simple Storage Service (Amazon S3)

B. Azure Blob storage

C. Amazon Elastic Block Store (Amazon EBS)

D. Azure Managed Disks

ANSWER: A B

Explanation:

Amazon Simple Storage Service (Amazon S3) is correct because ONTAP supports Amazon S3 object storage as a FabricPool cloud tier. ONTAP can tier eligible inactive data blocks from a local performance tier to the configured S3 bucket while retaining the active file system namespace on the ONTAP cluster.

Azure Blob storage is also correct because it is a supported FabricPool object-store provider. The ONTAP cluster requires appropriately configured connectivity, authentication, and bucket or container access to use the object store as a cloud tier. Amazon EBS and Azure Managed Disks are block-storage services rather than FabricPool object-store targets. See the ONTAP FabricPool documentation.

QUESTION NO: 8

You are creating a new meme generation application and are using containers for service chaining. The end user uploads the original image to one container, the image resolution modifier to another container, and the text overlay to a third container.

In this scenario, knowing that you have NetApp ONTAP storage available to you, which solution will allow each container to access the same data?

A. Re-encode the binary data into MIME64, and pass it as a string when instantiating new containers.

B. Presenting the same LUN to all of your nodes' underlying OS will make that storage available to the container as well.

C. Use the same container image for all three tasks and the data will exist because data in containers is persistent.

D. Provision a persistent volume using Trident, and then have Kubernetes claim this volume, as necessary, to provide the same storage to each container.

ANSWER: D

Explanation:

Provision a persistent volume using Trident, and then have Kubernetes claim this volume, as necessary, to provide the same storage to each container. This is the appropriate container-native approach because NetApp Trident integrates ONTAP with Kubernetes through the Container Storage Interface (CSI). A Kubernetes PersistentVolumeClaim can request storage from a Trident StorageClass, and Trident dynamically provisions a PersistentVolume on the ONTAP backend. Pods running the image-upload, image-processing, and text-overlay services can mount that persistent storage, allowing each stage of the workflow to read the source image and write its output in a shared, durable location.

For simultaneous access by separate containers or pods, the selected Trident backend and StorageClass must support an appropriate multi-node access mode, commonly ReadWriteMany for shared file storage. ONTAP NAS-backed Trident drivers, such as those using NFS, are designed for this kind of shared filesystem access. The data is therefore independent of individual container lifecycles and remains available as workload pods are restarted, replaced, or scaled. Kubernetes documents the relationship between PersistentVolumes, PersistentVolumeClaims, and pod volume mounts, while NetApp documents Trident's dynamic provisioning and ONTAP storage integration. See NetApp Trident for Kubernetes and Kubernetes Persistent Volumes.