PassQueen
a.k.a dumpsqueen.com

Cloud, Specialist (JNCIS-Cloud)

Juniper JN0-412

Version Demo

Total Demo Questions: 8

Total Premium Questions: 89

Buy Premium PDF

https://passqueen.com

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Cloud Concepts	10
Topic 2, Cloud Networking Concepts	35
Topic 3, Cloud Security	9
Topic 4, Cloud Automation and Orchestration	14
Topic 5, Cloud Operations	21
Total	89

QUESTION NO: 1

Which protocol does Contrail use to exchange information between the Contrail Controller and the Contrail vRouter?

A. NETCONF

B. XMPP

C. BGP

D. SSH

ANSWER: B

Explanation:

XMPP is correct. In Contrail, the control node in the Contrail Controller communicates routing and forwarding-control information to each Contrail vRouter using the Extensible Messaging and Presence Protocol (XMPP). The vRouter agent establishes an XMPP session to the control node, through which it receives information needed to program the local forwarding plane, including virtual-network reachability, routes, and related policy or next-hop information. The vRouter agent also uses this connection to advertise locally learned routes and operational state to the control node. This controller-to-vRouter relationship lets Contrail distribute overlay-network control-plane state dynamically while the vRouter performs packet forwarding on the compute host. Juniper's Contrail documentation describes XMPP as the protocol used between the control node and vRouter agent, and identifies the control node as the component that exchanges routes with vRouters. See the Juniper Contrail architecture overview and the Contrail Cloud architecture documentation.

QUESTION NO: 2

What is the proper order for creating a virtual network?

A. Configure the IPAM, virtual network, DNS, and then the virtual machine.

B. Configure the DNS, IPAM, virtual network, and then the virtual machine.

C. Configure the virtual network, DNS, IPAM, and then the virtual machine.

D. Configure the IPAM, DNS, virtual network, and then the virtual machine.

ANSWER: B

Explanation:

Configure the DNS, IPAM, virtual network, and then the virtual machine. is the required dependency order in Juniper Contrail Networking. A virtual DNS object is created first because it supplies the DNS configuration that an IP Address Management (IPAM) object can reference. The IPAM object then defines the address-management information for the network, including its subnets and the associated virtual DNS settings. Once that IPAM configuration exists, it can be associated with the virtual network while the virtual network is created. This gives the virtual network the IP allocation and name-resolution settings needed for attached workloads. Finally, create the virtual machine and attach its virtual-machine interface to the virtual network. At that point, Contrail can allocate an address from the configured IPAM subnet and apply the network's DNS-related configuration to the workload. Following this order ensures that every object is available before a later object needs to reference it, which is the workflow described in Juniper's virtual-network creation procedure. See Juniper's Creating a Virtual Network documentation and the Contrail Networking documentation.

QUESTION NO: 3

Which two functions are performed by the Contrail vRouter? (Choose two.)

A. Forward traffic for attached workloads.

B. Enforce distributed network policy.

C. Store OpenStack image files.

D. Provide identity tokens for cloud users.

ANSWER: A B

Explanation:

The Contrail vRouter performs packet forwarding for workloads attached to the compute host. It uses the routes, next hops, tunnel information, and forwarding state programmed by the vRouter agent to deliver traffic locally or across the overlay network.

The vRouter also enforces distributed network policy close to the workload interface. Policy rules can permit, deny, or redirect traffic before it leaves the compute host, which supports scalable segmentation without requiring all traffic to traverse a centralized firewall. See the Contrail architecture overview and the Juniper Documentation portal.

QUESTION NO: 4

In which OpenStack core service would you find disk snapshots?

A. Glance

B. Trove

C. Cinder

D. Swift

ANSWER: A

Explanation:

Glance is correct because it is OpenStack's Image service and is responsible for discovering, registering, retrieving, and storing virtual-machine images. In the instance lifecycle, a disk snapshot is commonly created by capturing an instance's current disk state as an image. That resulting snapshot image is then cataloged and made available through Glance, where it can be used later to launch a new instance with the captured operating-system and disk contents. Glance maintains image metadata, image locations, and APIs that other OpenStack services use to access these images. This role is especially important for cloud users who need to preserve a known-good server state, create reusable golden images, or duplicate an existing workload configuration. The OpenStack Image service documentation describes Glance as the service that provides image discovery, registration, and delivery capabilities, while the Compute service documentation explains that creating an image from an instance produces an image artifact for subsequent use. See the OpenStack Glance documentation and the OpenStack Compute image management guide.

QUESTION NO: 5

Which two nodes are part of the Contrail controller role? (Choose two.)

A. config

B. control

C. analytics

D. load balancer

ANSWER: A B

Explanation:

The Contrail controller role consists of the **config** and **control** nodes. The config node provides the configuration-plane services that accept and manage Contrail objects, such as virtual networks, policies, routing instances, and virtual-machine interfaces. It maintains the authoritative configuration state and distributes the required information to the other Contrail components.

The control node provides the network control plane. It establishes XMPP sessions with vRouters and uses BGP to exchange routing and reachability information with other control nodes and external routing devices. This allows the Contrail system to distribute routes and maintain forwarding information for virtualized workloads across the overlay network.

Together, these two node types form the essential controller-plane functions: configuration management and route/control-plane distribution. Juniper's Contrail architecture documentation describes the configuration and control services as distinct controller components and explains their respective responsibilities. See the Contrail architecture overview and Juniper's Contrail component documentation.

QUESTION NO: 6

Which two statements are true about BGP as a Service (BGPaaS) in Contrail? (Choose two.)

A. BGPaaS enables a workload to establish a BGP session with the Contrail control plane.

B. BGPaaS enables dynamic route advertisement by a tenant workload.

C. BGPaaS is a VXLAN encapsulation type.

D. BGPaaS replaces XMPP between vRouters and control nodes.

ANSWER: A B

Explanation:

BGPaaS allows a tenant workload, such as a virtual machine or network function, to establish a BGP session with the Contrail control plane. This enables the workload to advertise routes dynamically and receive routes without requiring static route configuration for each prefix.

BGPaaS is useful for workloads that run routing software, including virtual routers and network functions. It is not an overlay encapsulation and does not replace the XMPP sessions used between Contrail vRouters and control nodes. See the Juniper BGPaaS documentation.

QUESTION NO: 7

Which two statements are true about Generic Device Operations? (Choose two.)

A. Generic Device Operations execute multiple commands on multiple devices.

B. Generic Device Operations are executed using the REST API.

C. Generic Device Operations are executed using Ansible.

D. Generic Device Operations execute a single command on multiple devices.

ANSWER: C D

Explanation:

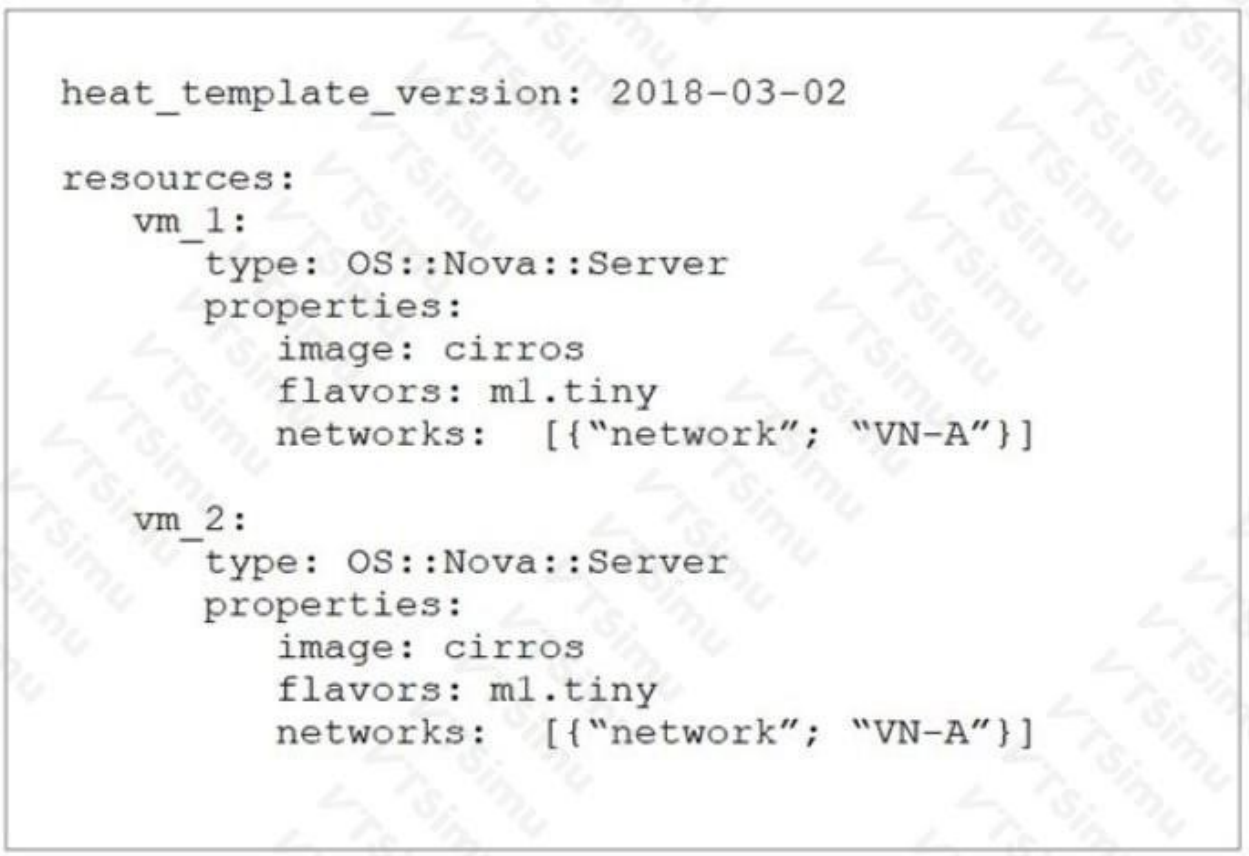
Generic Device Operations execute a single command on multiple devices. In Contrail, this capability is intended for operational, typically show-style, commands that can be sent to a selected group of managed physical devices in one operation. This provides a consistent way to collect the same operational information across devices without manually connecting to each device and issuing the command separately.

Generic Device Operations are executed using Ansible. Contrail uses Ansible as the automation mechanism for these operations, generating and running the required task against the devices selected for the operation. Ansible supplies the

orchestration layer that connects to the managed devices, invokes the requested operational command, and returns the gathered output through the Contrail workflow. This design makes Generic Device Operations useful for repeatable, multi-device operational verification while keeping the requested device action consistent for every target. Juniper documents the Generic Device Operational Commands workflow at Generic Device Operational Commands. Additional background on Ansible-based automation in Contrail is available in the Contrail Ansible Automation documentation.

QUESTION NO: 8

Click the Exhibit button.



```
heat_template_version: 2018-03-02

resources:
   vm_1:
      type: OS::Nova::Server
      properties:
         image: cirros
         flavors: m1.tiny
         networks:  [{"network"; "VN-A"}]

   vm_2:
      type: OS::Nova::Server
      properties:
         image: cirros
         flavors: m1.tiny
         networks:  [{"network"; "VN-A"}]
```

What will be the result of deploying the Heat template shown in the exhibit?

A. Neither VM will be deployed.

B. Only vm_2 will be deployed and vm_1 will be overwritten.

C. Both vm_1 and vm_2 will be deployed.

D. Only vm_1 will be deployed and vm_2 will fail.

ANSWER: C

Explanation:

Both vm_1 and vm_2 will be deployed. In a Heat Orchestration Template, each entry under the `resources` section is a separately managed logical resource. The template defines `vm_1` and `vm_2` as distinct Nova server resources, so Heat processes each resource declaration and requests creation of a corresponding server. Reusing an input parameter or using equivalent property values across resource definitions does not merge those resources or cause one logical resource to overwrite another. Resource identity within the stack is determined by the unique Heat logical resource names, which are `vm_1` and `vm_2` in this template. Heat therefore creates and tracks two independent `OS::Nova::Server` resources as part of the same stack deployment. The OpenStack Heat template guide describes resources as individually named entities in the `resources` section, while the Nova server resource reference documents the properties used when Heat creates a server. See the Heat Orchestration Template guide and the OS::Nova::Server resource reference.