Download Juniper JN0-224 Exam Dumps Free

Shared by Hughes on 17-06-2026

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

Exhibit.

```
<rpc-reply xmlns:junos="http://xml.juniper.net/junos/20.1R0/junos">
  <configuration junos:commit-seconds="1589295982" junos:commit-localtime="2020-05-12 15:06:22
UTC" junos:commit-user="user">
            <interfaces>
                  <interface>
                     <name>ge-0/0/1</name>
                     <unit>
                        <name>0</name>
                        <family>
                           <inet>
                              <address>
                                 <name>192.168.10.1/24</name>
                              </address>
                           </inet>
                        </family>
                     </unit>
                  </interface>
                  <interface>
                     <name>fxp0</name>
                     <unit>
                        <name>0</name>
                        <family>
                              <inet>
                                 <address>
                                 <name>172.25.11.1/24</name>
                              </address>
                           </inet>
                        </family>
                     </unit>
                  </interface>
            </interfaces>
   </configuration>
</rpc-reply>
```

Referring to the exhibit, which XML XPath expression will only show the IP address XML elements?

Options:

A- //address/name

B- //name

C- /name

D- /address/name

Answer:

A

Explanation:

https://www.w3schools.com/xml/xpath_syntax.asp

Question 2

Question Type: MultipleChoice

Given the following Python script:

a = [1,2,3,4,5,6,7,8,9]

print(a[0])

What is the output of this print command?

Options:

A- 1
B- 2
C- 7
D- 9

Answer:

A

Explanation:

In Python, lists are zero-indexed, meaning the first element of the list is at index 0. The given script is:

pythona = [1, 2, 3, 4, 5, 6, 7, 8, 9]

print(a[0])

a[0] refers to the first element in the list a, which is 1.

So, the output of the print(a[0]) command is 1.

Option A is correct because Python indexing starts at 0, making the first element of the list at index 0.

Python Official Documentation: Covers list indexing and operations.

Python Programming Tutorials: Provide examples of list indexing.

Question 3

Question Type: MultipleChoice

You are asked to develop an on-box Junos script that prevents deletion of the SNMP configuration.

Which type of script serves this purpose?

Options:

A- commit script
B- event script
C- op script
D- SNMP script

Answer:

A

Explanation:

A commit script in Junos is used to enforce policies and configuration constraints on the device. These scripts are written in Extensible Stylesheet Language Transformations (XSLT) or Python and are executed automatically during the commit process of a configuration change.

In this context, to prevent the deletion of the SNMP configuration, a commit script is the appropriate choice. It can be designed to check the configuration changes being committed and reject any commit that attempts to delete or modify the SNMP configuration. This script essentially acts as a gatekeeper, ensuring that only allowable changes are committed to the device configuration.

Supporting Reference:

Juniper Networks Commit Scripts Documentation: The official Juniper documentation provides examples and use cases of commit scripts, including how they can be used to prevent unauthorized changes to the device configuration.

'Junos Automation Scripting' by Jonathan Looney: This resource gives practical examples and best practices for creating commit scripts to enforce configuration policies in Junos OS.

Question 4

Question Type: MultipleChoice

You are asked to write an on-box script that will be triggered when a specific interface on a Junos device goes down.

Which type of on-box script should you use to accomplish this task?

Options:

A- commit
B- event
C- operation
D- SNMP

Answer:

B

Explanation:

An event script is used to automate responses to system events in Junos, such as an interface going down. These scripts are triggered automatically when a specified event occurs, making them suitable for tasks like monitoring interface status and executing actions when the status changes.

Option B (event) is correct because event scripts are designed for reacting to system events like an interface going down.

Option A (commit) is used for configuration changes, Option C (operation) is used for operational tasks, and Option D (SNMP) is not applicable in this context.

Supporting Reference:

Juniper Networks Event Scripts Documentation: Details how event scripts are used to automate responses to specific system events in Junos

Question 5

Question Type: MultipleChoice

Which two statements are correct about using the Junos REST API? (Choose two.)

Options:

A- It supports data In CSV format.
B- It must use SSH for a connection.
C- NETCONF is not supported.
D- It is a simple configuration.

Answer:

A, D

Explanation:

A. It supports data in CSV format: The Junos REST API supports multiple data formats for transferring information between systems, including XML, JSON, and CSV (Comma Separated Values). This flexibility allows for easier data parsing, especially in environments where structured data (like CSV) is a standard. CSV is often used for bulk data export or import and reporting purposes, making it an essential format for automation tasks involving external systems or large datasets.

Example Usage in REST API: When using the Junos REST API, a user can request configuration or operational data and specify the response format (XML, JSON, or CSV). CSV is particularly useful when integrating Junos devices with systems that require easily readable, tabular formats.

D. It is a simple configuration: The Junos REST API is designed to be relatively simple to configure. Once the REST API service is enabled on the Junos device, it can be accessed via HTTP or HTTPS, making it an easy entry point for automation and management tasks. Unlike more complex protocols (such as NETCONF), the REST API is lightweight and easier to use for simple configuration changes and retrieving operational data.

Configuration Example:

To enable the REST API, you can add the following configuration:

set system services rest http

set system services rest https

After enabling the service, API requests can be made to interact with the device for automation tasks, without needing the complexity of SSH or NETCONF configuration.

Why the Other Options Are Incorrect:

B . It must use SSH for a connection: This is incorrect. The Junos REST API uses HTTP or HTTPS for communication, not SSH. While SSH is commonly used for NETCONF, it is not required for REST API connections. REST APIs operate over standard web protocols.

C . NETCONF is not supported: This is incorrect. Junos supports both REST API and NETCONF for automation and configuration management. NETCONF is an XML-based protocol used for device configuration, which operates over SSH. The REST API and NETCONF can coexist on the same device, offering multiple avenues for automation.

Juniper Automation in DevOps Context: The simplicity and flexibility of the Junos REST API make it ideal for DevOps automation tasks. It allows teams to easily interact with Junos devices using lightweight RESTful methods, integrating with external systems through formats like CSV. The ease of configuration supports rapid deployment and scaling of automated management tasks.

Reference from Juniper Documentation:

Junos REST API Documentation

Question 6

Question Type: MultipleChoice

Which two statements are valid regarding Junos automation? (Choose two.)

Options:

A- The jsd process handles XML API calls.
B- The mgd process handles JET API requests.
C- The jsd process handles JET API requests.
D- The mod process handles XML API calls.

Answer:

A, C

Explanation:

In Junos automation, several processes handle API requests, and understanding which process handles what is crucial:

jsd Process:

XML API Calls (A): The jsd process is responsible for handling XML API calls, which are a significant part of Junos automation. XML API allows for structured and standardized communication with

Junos devices, enabling automation scripts to query and configure devices.

JET API Requests (C): The jsd process also handles JET (Junos Extension Toolkit) API requests. JET provides a more modern, programmable interface for interacting with Junos OS, and jsd is the process that manages these interactions.

mgd Process (Incorrect Option):

Not for JET API Requests: The mgd process handles general management operations, such as CLI commands and managing the configuration database, but it does not handle JET API requests. That role is fulfilled by jsd.

mod Process (Incorrect Option):

Not for XML API Calls: The mod process deals with managing chassis components and is not involved in handling XML API calls.

Juniper Networks JET and XML API Documentation: Describes the roles of jsd in handling both XML and JET API requests.

Junos Automation and DevOps Documentation: Provides insights into how different processes interact with Junos APIs.

Question 7

Question Type: MultipleChoice

Which two statements about NETCONF are true? (Select two.)

Options:

A- It uses the operations layer to lock the configuration of a Junos device.
B- It uses the messages layer to commit the configuration of a Junos device.
C- It uses the messages layer to lock the configuration of a Junos device.
D- It uses the operations layer to commit the configuration of a Junos device.

Answer:

A, D

Explanation:

NETCONF (Network Configuration Protocol) operates through different layers, with the operations

layer being particularly important for managing configurations:

Operations Layer (A & D): This layer is responsible for actions like locking and committing the configuration on a Junos device. The lock operation prevents other sessions from modifying the configuration, and the commit operation applies the configuration changes to the device.

Options B and C are incorrect because the messages layer handles the communication aspects, such as exchanging data between the client and server, not performing configuration operations like locking and committing.

IETF RFC 6241 (NETCONF): Describes the protocol layers and their functions, with a focus on the operations layer.

Juniper Networks NETCONF Documentation: Provides insights into how NETCONF operations are managed in Junos

Question 8

Question Type: MultipleChoice

Which data construct is used to guarantee that element names and data values remain unique in an XML document?

Options:

A- element
B- schema definition
C- namespace
D- dictionary

Answer:

C

Explanation:

In XML documents, a namespace is the data construct used to ensure that element names and data values remain unique. Namespaces prevent naming conflicts by differentiating between elements or attributes that may have the same name but different meanings. This is particularly important in XML, where documents often incorporate elements from multiple sources.

Detailed Explanation:

XML Namespaces: A namespace is a collection of names, identified by a URI reference, which is used to distinguish between elements that may have identical names but different definitions or origins. This helps avoid ambiguity in the document.

How Namespaces Work: When a namespace is applied, each element or attribute in the XML document is associated with a prefix. This prefix, combined with the namespace URI, ensures that the element or attribute is uniquely identified, even if another element or attribute in the same document has the same local name but a different namespace.

Schema Definition vs. Namespace: Although an XML schema definition (XSD) can define the structure and type constraints of an XML document, it does not guarantee uniqueness of element names across different XML documents. That role is fulfilled by namespaces.

Practical Example:

xml

Copy code

<root xmlns:ns1='http://www.example.com/ns1'

xmlns:ns2='http://www.example.com/ns2'>

<ns1:item>Item in namespace 1</ns1:item>

<ns2:item>Item in namespace 2</ns2:item>

</root>

In this example, the item elements are in different namespaces (ns1 and ns2), which keeps them unique even though they have the same name.

Juniper Automation and DevOps Documentation: These practices highlight the importance of namespaces in XML documents to maintain the integrity and uniqueness of data, which is essential in automation scripts and configuration files.

W3C XML Namespace Specification: The World Wide Web Consortium (W3C) standard for XML Namespaces defines how namespaces should be used to avoid name conflicts.

Namespaces are a crucial concept in XML, ensuring that data can be consistently managed and interpreted correctly, particularly in complex systems where multiple XML documents or schemas are involved.

Question 9

Question Type: MultipleChoice

Exhibit.

```
#!/usr/bin/python

from jnpr.junos import Device
from pprint import pprint
from lxml import etree

  with Device("172.25.11.1", user="lab", passwd="lab123") as dev:
      config_data = dev.rpc.get_config(filter_xml="interfaces",
options={"format": "json"})

      print("Configured interfaces:")
      for interface in config_data["configuration"]["interfaces"]
["interface"]:
          print(interface["name"])
```

Referring to the exhibit, which two statements about the script are correct? (Choose two.)

Options:

A- The script prints the name of each configured interface.
B- The script retrieves the interface configuration in XML
C- The script prints interface information for each interface name.
D- The script retrieves the interface configuration in JSON.

Answer:

A, D

Question 10

Question Type: MultipleChoice

Which two programming languages are used for Junos on-box scripting? (Select two.)

Options:

A- Perl
B- Ruby
C- SLAX
D- XSLT

Answer:

C, D

Explanation:

Junos on-box scripting supports the following programming languages:

SLAX (C): SLAX (Structured Language for XML) is a scripting language designed specifically for Junos devices. It allows for easy manipulation of XML data, making it ideal for creating Junos scripts that interact with device configurations.

XSLT (D): XSLT (Extensible Stylesheet Language Transformations) is another language used for transforming XML documents into other formats. It is commonly used in Junos for transforming XML data into different views or outputs.

Options A (Perl) and B (Ruby) are not used for Junos on-box scripting. While these languages are popular in other contexts, Junos scripting relies heavily on XML-based languages like SLAX and XSLT.

Junos XML API and Scripting Guide: Describes the use of SLAX and XSLT for on-box scripting.

Juniper Networks Automation Documentation: Provides examples and best practices for using SLAX and XSLT in Junos scripting.

Question 11

Question Type: MultipleChoice

Which statement about the NETCONF content layer is true?

Options:

A- It uses YAML for RPC request and response payloads.
B- It uses XML for RPC request and response payloads.
C- It uses JSON for RPC request and response payloads.
D- It uses HTML for RPC request and response payloads.

Answer:

B

Explanation:

The NETCONF protocol, used for network management, utilizes XML for encoding the RPC (Remote Procedure Call) requests and responses. XML is chosen because of its flexibility and ability to represent hierarchical data structures, making it well-suited for representing network configurations and states.

Option B is correct because XML is the standard format used for NETCONF RPC payloads.

Options A (YAML), C (JSON), and D (HTML) are incorrect because these formats are not used by NETCONF for its RPC payloads.

Supporting Reference:

RFC 6241 - NETCONF Protocol: This RFC describes the use of XML for encoding NETCONF messages.