

DUMPSBOSS.

Nokia NSP IP Network Automation Professional Composite Exam

Nokia 4A0-AI1

Version Demo

Total Demo Questions: 9

Total Premium Questions: 98

Buy Premium PDF

<https://dumpsboss.co>

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

QUESTION NO: 1

Which of the following is NOT supported by the Workflow Manager out of the box?

- A. REST/RESTCONF APIs
- B. Optical integration using TL1
- C. Shell command execution
- D. CLI command execution on SROS and third party routers

ANSWER: C

Explanation:

Shell command execution is the capability that is not supplied as an out-of-the-box Workflow Manager integration. Nokia NSP Workflow Manager is designed to orchestrate network and service automation through its packaged workflow actions and supported adapters. These actions provide direct, controlled interaction with network-management and network-element interfaces, including REST/RESTCONF APIs, router CLI operations, and optical-system integration through TL1. This model lets workflows invoke supported management interfaces while retaining workflow visibility, credentials handling, execution tracking, and operational controls within NSP.

Although the underlying workflow platform can be extended and workflows can invoke external systems through supported interfaces, arbitrary execution of operating-system shell commands is not a standard packaged Workflow Manager capability. Implementing such a requirement would normally require a purpose-built integration, external automation service, or a custom action designed and governed for the deployment. Nokia documents Workflow Manager as the NSP component for creating, executing, and monitoring automated workflows; its available integrations and actions depend on the installed NSP release and associated documentation. See the [Nokia NSP documentation portal](#) and the [OpenStack Mistral documentation](#) for workflow-engine background and supported action-extension concepts.

QUESTION NO: 2

Which of the following NSP CLI actions uses the network element's NE ID to establish communication?

- A. nsp.cli
- B. nsp.sr_cli
- C. nsp.mdm_di
- D. nsp.netconf

ANSWER: B

Explanation:

`nsp.sr_cli` is the NSP workflow action intended for nodal CLI communication with an SR OS network element. Its connection context is based on the network element record known to NSP, identified through the NE ID. This lets NSP resolve the managed node and use its established management information and credentials rather than requiring the workflow designer to construct a direct device session independently. The action is therefore appropriate when a workflow needs to send CLI commands to a specific managed SR OS router and collect the command output as part of an automation process. Using the NE ID also aligns the workflow with NSP's inventory and device-management model: the workflow addresses the same network element object that NSP manages, allowing automation to remain tied to the authoritative managed-device identity. Nokia's NSP documentation groups these capabilities under Network Automation workflow actions and describes managed network elements as inventory objects used by automation services. See the [Nokia NSP documentation portal](#) for the relevant release documentation and the [Nokia Network Services Platform overview](#) for the platform's managed-network automation context.

QUESTION NO: 3

Which directory or directories do the file actions provide access to on the NSP server?

- A. /tmp and /mnt
- B. all directories
- C. /home
- D. /opt

ANSWER: B

Explanation:

The correct answer is **all directories**. NSP file actions are designed to work with files and folders on the NSP server rather than being limited to a predefined set of paths such as temporary, mount, home, or application directories. Their effective access is governed by the operating-system permissions associated with the user account that executes the action. Consequently, a file action can target any directory path on the server when that executing user has the necessary read, write, or execute permissions for the relevant directory and file. This follows the standard Linux discretionary-access-control model: directory permissions control traversal and listing, while file permissions control operations such as reading, modifying, or deleting a file. In practice, automation designers should use an appropriately permissioned service account and grant only the access required by the workflow, particularly when actions handle configuration exports, generated reports, backups, or uploaded artifacts. Nokia's NSP documentation portal is available at [Nokia NSP Documentation](#). For background on Linux file and directory permission behavior that determines the file action's effective access, see [GNU Coreutils: file permission information](#).

QUESTION NO: 4

Which of the following is NOT an NSP application?

- A. Service Fulfilment
- B. Network Supervision
- C. IP/MPLS Optimization
- D. Network Virtualization

ANSWER: D

Explanation:

Network Virtualization is the correct choice because it is not the name of an NSP application in Nokia's NSP application portfolio. Nokia NSP is an automation and management platform whose named applications include capabilities such as Service Fulfilment, Network Supervision, and IP/MPLS Optimization. Those applications address the operational lifecycle of IP networks: provisioning and service activation, assurance and supervision, and traffic-engineering optimization.

Network virtualization can certainly be a network capability or architectural concept managed through automation platforms, including through overlay, segmentation, and cloud-related workflows. However, that concept should not be confused with a separately branded NSP application called "Network Virtualization." Nokia's product material describes NSP as providing intent-based automation, service lifecycle management, network assurance, and optimization across multi-vendor IP and optical environments; it does not identify Network Virtualization as one of the NSP applications. This distinction matters in certification questions because product application names are specific labels, while virtualization is a broader technology domain. See Nokia's [Network Services Platform product page](#) and the [Nokia NSP documentation portal](#).

QUESTION NO: 5

Which of the following is NOT a benefit of Service Fulfilment API?

- A. Increases service agility

- B. Reduces human errors
- C. Allows multi-vendor service provisioning
- D. Provides alarm monitoring

ANSWER: D

Explanation:

Provides alarm monitoring is the correct answer because the Service Fulfilment API is intended for programmatic service fulfilment: it exposes capabilities that let external systems request, automate, and manage service provisioning workflows through NSP. Its value is in making service delivery more repeatable and API-driven, enabling orchestration systems or customer-facing portals to initiate fulfilment without relying on manual, device-by-device operational steps. This supports faster delivery of services and more consistent execution of provisioning processes.

Alarm monitoring belongs to the assurance and fault-management domain rather than the fulfilment domain. In NSP, alarm-related capabilities are associated with network monitoring, event handling, and assurance functions, which are separate from the API's core service-order and provisioning purpose. Therefore, alarm monitoring is not a stated service-fulfilment benefit. Nokia's NSP documentation distinguishes service fulfilment and assurance capabilities across its product documentation; see the [Nokia NSP documentation portal](#) and the [Nokia Network Services Platform overview](#).

QUESTION NO: 6

Which of the following is NOT a valid flow control attribute?

- A. on-failure
- B. on-complete
- C. on-success
- D. on-error

ANSWER: A

Explanation:

`on-failure` is not a valid NSP flow-control attribute. In NSP workflow definitions, flow control uses the defined outcome hooks to determine what execution path follows a task result. The supported naming convention includes hooks for successful completion, error handling, and completion processing; it does not define a separate `on-failure` hook. Consequently, using `on-failure` in a workflow definition does not express a recognized transition condition and cannot be relied upon to route workflow execution after a task fails. Workflow authors must use the documented flow-control vocabulary exactly, because these attributes are interpreted by the workflow engine rather than treated as arbitrary labels. Nokia's NSP documentation portal is the authoritative source for the release-specific workflow and automation guides, where the available workflow syntax and attributes should be checked for the deployed NSP version. See the [Nokia NSP documentation portal](#) and Nokia's [Network Services Platform product page](#) for product and documentation access.

QUESTION NO: 7

Which of the following is NOT a characteristic of Infrastructure Iure Intents?

- A. Policy management
- B. System security
- C. Initial hardware configuration
- D. L2/L3 VPN configuration of multiple services

ANSWER: D

Explanation:

L2/L3 VPN configuration of multiple services is not a characteristic of infrastructure lifecycle intents. In Nokia NSP Intent Manager, infrastructure-oriented intents are used to establish, maintain, and govern the underlying network infrastructure on which services operate. Their scope includes foundational operational concerns such as initial device or hardware configuration, policy control, and security-related configuration. These intents help ensure that network elements and common infrastructure capabilities are consistently prepared and managed throughout their lifecycle.

Configuring Layer 2 or Layer 3 VPNs for multiple services has a different purpose: it defines and provisions customer or application-facing connectivity services. Such work is normally modeled through service-oriented intent workflows, which express the desired service topology, endpoints, and service parameters, then validate and deploy the required network configuration. Separating infrastructure lifecycle management from service provisioning allows NSP automation to manage reusable platform capabilities independently from the individual VPN services delivered over that platform. Nokia's Intent Manager documentation describes intent-based automation concepts and lifecycle management for network configuration, while the NSP documentation portal provides the product documentation set: [Nokia NSP Intent Manager documentation](#) and [Nokia NSP documentation portal](#).

QUESTION NO: 8

An external OSS needs the service status and administrative state for a selected set of intents. The OSS does not need the remaining fields returned by a broad resource representation and wants to request only the required data in one query. Which NSP API style is best suited to this requirement?

- A. SOAP/XML
- B. SNMP
- C. GraphQL
- D. RESTCONF

ANSWER: C

Explanation:

GraphQL is correct because its query model enables an API client to request the specific fields it requires. This is useful when an OSS needs a focused view of multiple related data items without consuming a larger fixed representation.

RESTCONF remains appropriate for accessing and modifying YANG-modeled resources, but the scenario specifically emphasizes selecting the precise returned fields through a query-oriented interface. SOAP/XML is a different service-integration style, while SNMP is a network-management protocol rather than an NSP API style for this requirement.

QUESTION NO: 9

Which of the following is NOT an Intent Type life-cycle state?

- A. Draft
- B. Released
- C. Phased-Out
- D. Created

ANSWER: D

Explanation:

Created is correct because it is not one of the controlled life-cycle states assigned to an Intent Type in Nokia NSP Intent Manager. Intent Type life-cycle management is used to govern the availability and evolution of the intent definition itself. The relevant states support the normal progression from development, through operational availability, to withdrawal from use. This governance allows NSP to manage which version of an Intent Type can be used for intent creation and supports safe evolution of automation content without disrupting deployed services.

“Created” describes an occurrence or action—an object having been created—rather than a defined Intent Type life-cycle classification. It therefore does not express the Intent Type’s governed maturity or availability within the Intent Manager catalog. When working with Intent Types, engineers should distinguish between an Intent Type’s life-cycle state and the operational states or events associated with individual intent instances. Nokia’s NSP documentation portal provides the Intent Manager guides and release-specific lifecycle details at [Nokia NSP Documentation](#). Additional NSP product context, including its automation and lifecycle-management role, is available at [Nokia Network Services Platform](#).