

# Release Notes

Published  
2025-07-22

## Juniper Paragon Automation Release 2.2.0

---

### Software Highlights

- **Device Life-Cycle Management**—Use IPv6 addresses to onboard and manage devices. Reboot Cisco devices, and configure device bandwidth and access parameters in the network implementation plan.
- **Observability**—View changes in topology in real-time and monitor overall network health using the WAN Health dashboard.
- **Trust and Compliance**—Use the Trust dashboard to view the overall trust score and monitor, in real-time, vulnerabilities that affect targets in the network (Beta feature).
- **Service Orchestration**—Use **Update Placements** to automatically allocate network resources for provisioning L3VPN and EVPN services (Beta feature).
- **Active Assurance**—Register Test Agent Applications, associate a Test Agent to a site, and view Monitors and Test Agents events.
- **Administration**—Enable RADIUS authentication to authenticate and authorize devices during onboarding.
- **Installation and Upgrade**—Configure the Paragon Automation cluster using IPv6 addresses and upgrade to Juniper Paragon Automation Release 2.2.0 from Release 2.1.0.

# Table of Contents

**Introduction | 1**

**Licensing | 2**

**Supported Junos OS Releases, Devices, and Browsers | 3**

**New Features | 5**

**Known Issues | 12**

**Resolved Issues | 30**

# Introduction

Service providers, cloud providers, and enterprises are facing an increase in the volume, velocity, and types of traffic. This creates both unique challenges (increased user expectations and expanded security threats) and fresh opportunities (new generation of 5G, IoT, distributed edge services) for network operators.

To accommodate rapid changes in traffic patterns, service providers and enterprises need to quickly detect and troubleshoot devices and service issues, and make changes to service configurations in real-time. Any misconfiguration due to human errors can lead to service outages. Investigating and resolving these issues can be a time-consuming process.

Juniper® Paragon Automation is a WAN automation solution that enables service provider and enterprise networks to meet these challenges. Juniper's solution delivers an experience-first and automation-driven network that provides a high-quality experience to network operators.

Paragon Automation is based on a modern microservices architecture with open APIs. Paragon Automation is designed with an easy to use UI that provides a superior operational and user experience. For example, Paragon Automation implements different persona profiles (such as network architect, network planner, field technician, and Network Operations Center [NOC] engineer) to enable operators to understand and perform the different activities in the device life-cycle management (LCM) process.

Paragon Automation takes a use case-based approach to network operations. When you execute a use case, Paragon Automation invokes all the required capabilities of that use case, runs a workflow (if necessary) and presents you with a completed set of tasks that implements the use case.

Paragon Automation supports the following use cases:

- **Device life-cycle management (LCM)**—Allows you to onboard, provision, and then manage a device. Paragon Automation automates the device onboarding experience, from shipment through service provisioning, thus enabling the device to be ready to accept production traffic.
- **Observability**—Allows you to visualize the network topology, view topology updates in real-time, and monitor devices and the network. You can also view device and network health and drill down into the details. In addition, Paragon Automation notifies you about network issues using alerts, alarms and events, which you can use to troubleshoot issues affecting your network.
- **Trust and compliance**—Automatically checks whether the device complies with the rules defined in the Center for Internet Security (CIS) benchmarks document. In addition, Paragon Automation also checks the configuration, integrity, and performance of the device and then generates a trust score that determines the device's trustworthiness.
- **Service Orchestration**—Enables you to streamline and optimize the delivery of network services, thereby improving efficiency and reducing the risk of errors. A service can be any point-to-point, point-to-multipoint or multipoint-to-multipoint connection. For example, Layer 3 VPNs or EVPNs.

- **Active Assurance**—Enables you to actively monitor and test the network's data plane by generating synthetic traffic using Test Agents. Test Agents are measurement points deployed in certain routers in your network. In addition, there are pre-deployed Test Agents in certain clouds, such as AWS. These Test Agents are capable of generating, receiving, and analyzing network traffic and therefore enable you to continuously view and monitor both real-time and aggregated result metrics.

For details about these use cases and other features of Paragon Automation, see ["New Features" on page 5](#).

In summary, Paragon Automation helps operators to automate the onboarding and provisioning of devices, simplify and accelerate service delivery, evaluate device and service performance, and reduce manual effort and timelines.

Use these release notes to know about features, supported Junos OS and Junos OS Evolved releases, supported devices, and open issues in Paragon Automation.

## Licensing

To use Paragon Automation and its features, you need:

- **Product Entitlement**—To use Paragon Automation and its use cases.



**NOTE:** Product entitlements are honor-based and not enforced for Paragon Automation Release 2.2.0.

- **Device License**—To use the features on a device that you onboarded.

To purchase a license, contact your [Juniper Networks](#) sales representative. For more information about purchasing licenses, see [Juniper Licensing User Guide](#). After you purchase a license, you can download the license file and manage licenses by using the [Juniper Agile Licensing](#) (JAL) portal. You can also choose to receive the license file over an e-mail. The license file contains the license key. The license key determines whether you are eligible to use the licensed features.

After the device is onboarded, the Super User and the Network Admin can add a device license from the **Licenses** tab (**Observability** > **Health** > **Troubleshoot Devices** > *Device-Name* > **Inventory** > **Licenses**) of the Paragon Automation GUI. For more information, see [Manage Device Licenses](#).

# Supported Junos OS Releases, Devices, and Browsers

[Table 1 on page 3](#) lists the supported Junos OS release, devices, and browsers in Juniper Paragon Automation.

**Table 1: Supported Junos OS release, devices, and browsers**

|                    |                                                                                                                                                                                 |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Supported Junos OS | <ul style="list-style-type: none"><li>• Junos OS Evolved releases 23.2R2, 22.4R2, 22.2R3, and 23.4R2.</li><li>• Junos OS releases 23.2R2, 22.4R2, 22.2R3, and 23.4R2.</li></ul> |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

**Supported Juniper Devices**

- ACX7024
- ACX7024-X
- ACX7100-32C
- ACX7100-48L
- ACX7348
- ACX7332
- ACX7509
- PTX10001-36MR
- PTX10004
- PTX10008
- PTX10016
- MX204
- MX240
- MX304
- MX480
- MX960
- MX10003
- MX10004
- MX10008
- vMX

|                                      |                                                                                                                                                                                                                                                                                                            |
|--------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Supported third-party devices</b> | <ul style="list-style-type: none"> <li>• Cisco Network Convergence System 57C3 (Cisco NCS57C3)</li> <li>• Cisco Network Convergence System 5504 (Cisco NCS5504)</li> <li>• Cisco 8202 Router</li> <li>• Cisco IOS XRv Router</li> <li>• Cisco Aggregation Services Routers 9902 (Cisco ASR9902)</li> </ul> |
| <b>Supported Browsers</b>            | The latest version of Google Chrome, Mozilla Firefox, and Safari.                                                                                                                                                                                                                                          |

## New Features

### IN THIS SECTION

- [Device Life-Cycle Management | 6](#)
- [Observability | 6](#)
- [Trust and Compliance | 7](#)
- [Service Orchestration | 8](#)
- [Active Assurance | 9](#)
- [Administration | 10](#)
- [Installation and Upgrade | 10](#)
- [Beta Features | 11](#)

This section describes the features available in Juniper Paragon Automation Release 2.2.0.

# Device Life-Cycle Management

Device life-cycle management (LCM) encompasses the entire life-cycle of the device, from installing the device on-site, bringing the device under management, monitoring the device when it is in production, and finally decommissioning the device.

Juniper Paragon Automation Release 2.2.0 provides the following additional device life-cycle management features:

- **Use IPv6 to onboard and manage devices**—Paragon Automation supports the use of IPv6 addresses to onboard and manage devices. In earlier releases, Paragon Automation supported only IPv4 addresses.

Paragon Automation does not support dual stack IP addressing in Release 2.2.0. You can use either IPv4 addresses only or IPv6 addresses only and cannot use a combination of IPv4 and IPv6 addresses for onboarding and managing devices.

- **Reboot a Cisco device**—You can use the **Reboot** option on the Troubleshooting Devices page (**Observability > Troubleshooting Devices**) to reboot Cisco devices.

[See [About the Troubleshooting Devices Page](#).]

## Observability

Paragon Automation enables you to view your entire network topology in real time, monitor network health, be notified of any anomalies in the network, and also get guidance on the remediation of these anomalies. With observability, Paragon Automation monitors and analyzes the network and its components by using key performance indicators (KPIs), device logs, and metrics, and notifies you about network issues through alerts and alarms. Additionally, Paragon Automation runs connectivity tests using synthetic traffic to identify connection issues between devices in your network. The timely detection of anomalies enables you to take prompt action and minimize the impact of any issues that occur.

Juniper Paragon Automation Release 2.2.0 provides the following additional observability features:

- **Support for dynamic topology**—You can view changes in topology in real time. Paragon Automation establishes a BGP-LS peering session with the devices in your network, and enables you to view changes in the network in real time even if the devices are not managed by Paragon Automation.



By default, the dynamic topology feature is disabled. To enable dynamic topology, you must specify the IP address of the BGP-LS peer and autonomous system (AS) number on the Topology page (**Observability > Network > Topology**).

[See [Dynamic Topology Workflow](#).]

- **Monitor overall network health**—Paragon Automation provides a dashboard that enables you to monitor network health-related data in real time. The WAN Health tab (**Observability > Health > Health Dashboard > WAN Health**) of the dashboard provides a unified view of the health of devices, interfaces, and routing components in the network. You use this information to perform corrective actions when necessary. You can view the following accordions on the WAN Health tab:
  - **Devices Health Accordion**—View the percentage of healthy devices and the total number of unhealthy devices, and a graph that displays the average health of all devices. You can also view the health of individual device chassis components and temperature.
  - **Interfaces Health Accordion**—View the percentage of healthy devices and the total number of unhealthy devices based on interface health, and a graph that displays the average health of all interfaces. You can also view details of link failure, input traffic, output traffic, and errors that affect the health of interfaces.
  - **Routing Health Accordion**—View routing alerts, the percentage of healthy devices, the total number of unhealthy devices, and a graph that displays the overall routing health. You can also view the health of individual routing components.

Click **View Details** on any accordion to view the related health pages where you can view the percentage of healthy devices and a list of KPIs. You can also view the total number of unhealthy devices grouped by OS version, device model, and sites.

[See [About the WAN Health Tab](#).]

- **View monitored pluggables and interfaces**—Paragon Automation monitors the health of device interfaces during device onboarding and when the device is operational. In this release, you can also view the total number of monitored pluggables and interfaces on the Interfaces accordion.

[See [Interfaces Data and Test Results](#).]

## Trust and Compliance

Paragon Automation helps protect the network from threats and vulnerabilities by periodically checking whether a target's configuration, integrity, and performance comply with predefined security benchmarks. The term target refers to devices and device components. Paragon Automation distills the

outcomes of these checks into a single trust score that you can use to determine how trustworthy a device is.

There are no new trust and compliance features in Juniper Paragon Automation Release 2.2.0. See "[Beta Features](#)" on [page 11](#) for information on features with Beta support in this release.

## Service Orchestration

Service orchestration is the process of designing, configuring, validating, deploying, and monitoring a network service. Paragon Automation automates the entire life cycle of a network service by providing workflows that execute the tasks to be completed to deliver a service. You can provision various network services by using predefined service designs written in YANG. The Service Catalog is an inventory of service designs, which are templates that provide guidelines and parameters for instantiating a service. A service instance defines the elements of a service. A service order includes the instruction to create, modify, or delete a service instance. After you initiate a service order and provision it, Paragon Automation activates the automated workflow to provision the service in the network. After provisioning, Paragon Automation monitors the service by automatically setting up Juniper® Paragon Insights and Juniper® Paragon Active Assurance instances to monitor network health and measure service quality.

Juniper Paragon Automation Release 2.2.0 provides the following additional service orchestration features:

- **Schedule service order provisioning**—You can schedule provisioning of an EVPN, L3VPN, and L2 circuit service order by specifying the date and time for provisioning. Paragon Automation automatically provisions the service order on the specified date and time.

[See [Add an L3VPN Service Instance](#), [Add an EVPN Service Instance](#), and [Add an L2 Circuit Service Instance](#).]

- **Monitor health of EVPN and L3VPN services**—Paragon Automation automatically monitors service health and quality after provisioning the service in the network. You can view health of monitoring data for EVPN and L3VPN services and their related components under the Passive Assurance tab (**Orchestration > Instances > Service Instance > *service-instance-name* hyperlink > Passive Assurance**).

[See [View Passive Assurance Monitoring Data](#).]

# Active Assurance

Active Assurance is a programmable test and monitoring solution, which generates synthetic traffic in the underlay network to gain continuous insights on network quality, availability, and performance. Active Assurance uses Test Agents, which are measurement points in your network. Test Agents generate and receive synthetic traffic, and enable you to continuously monitor and validate the infrastructure. You can deploy the Test Agents at strategic locations in your network and install them on Junos OS Evolved routers, x86 hardware, or on virtual machines. If you are using Juniper Networks® MX Series Universal Routers and Juniper Networks® PTX Series Routers, Paragon Automation uses real-time performance monitoring (RPM) for collecting the metric data.

Juniper Paragon Automation Release 2.2.0 provides the following additional Active Assurance features:

- **Register Test Agent Applications**—Paragon Automation can discover a Test Agent Application in your network.

For Paragon Automation to discover a Test Agent Application, you must register the Test Agent name and tags, if any, on the Add Test Agent page (**Inventory > Active Assurance > Test Agents > Add Test Agent**). After you add a Test Agent Application, Paragon Automation generates commands and a secret key that you can view and copy from the Test Agent Details page (**Inventory > Active Assurance > Test Agents > Test Agent Details**). When you execute these generated commands in the Docker environment, Paragon Automation discovers the Test Agent Application and adds this Test Agent Application to the Test Agent inventory list.

[See [Add a Test Agent](#).]

- **Identify the location of a Test Agent in your network**—Paragon Automation enables you to assign a Test Agent to a specific site from the Test Agent Details page (**Inventory > Active Assurance > Test Agents > Test Agent Details**). When you assign a site to a Test Agent, you can easily identify the location of the Test Agent.

[See [About the Test Agent Details Page](#).]

You can assign a Test Agent to a site only if the Test Agent is not installed on any devices.

- **View events related to Test Agents and Monitors**— In the Events column, you can view the number of events generated for Test Agents on the Test Agents page (**Inventory > Active Assurance > Test Agents**) and Monitors on the Monitors page (**Observability > Active Assurance > Monitors**).

Click an *Event-Number* link in the Events column to view events and their details (severity level, description, raise time, and clear time) on the Events page.

[See [About the Test Agents Page](#) and [About the Monitors Page](#).]

# Administration

Paragon Automation Release 2.2.0 provides the following administration features to manage users, sites, and organizations:

- **Support for RADIUS authentication and authorization of devices during onboarding**—Paragon Automation supports authentication and authorization of devices by using RADIUS during onboarding a device to Paragon Automation. To use RADIUS authentication, you must enable RADIUS under **System Settings** and configure at least one RADIUS server in Paragon Automation.

You do not need to be a superuser or a network administrator to onboard a device using RADIUS authentication.

[See [Manage RADIUS Server Configuration](#).]

- **Logging device configuration details**—When a service is provisioned, Paragon Automation logs the following details:
  - Service design and its version
  - User who created the service order
  - ID of the service instance related to the service design
  - Details of the service order workflow such as the time the workflow was initiated, and the workflow run ID
  - ID of the device that was configured by the workflow
  - Command sent to the device
  - Device's response

[See [Audit Logs Overview](#).]

## Installation and Upgrade

Juniper Paragon Automation Release 2.2.0 provides the following installation features:

- **Upgrade support**—You can upgrade your existing Juniper Paragon Automation Release 2.1.0 cluster to Release 2.2.0.

[See [Upgrade Paragon Automation](#).]

- **Support for IPv6 addresses**—You can configure the Paragon Automation cluster using IPv6 addresses in addition to IPv4 addresses. Paragon Automation supports the use of IPv6 addresses for the following:

- Cluster node virtual machines (VMs)
- Generic common ingress (gNMI, OC-TERM, and the Web GUI)
- Active Assurance Test Agent gateway

To configure IPv6 addresses, you must install Juniper Paragon Automation Release 2.2.0 afresh. You cannot configure IPv6 on a setup upgraded from Release 2.1.0.

[See [Paragon Automation System Requirements](#) and [Install Paragon Automation](#).]

## Beta Features

Juniper Paragon Automation Release 2.2.0 provides Beta support for the following features:

- **Configure device bandwidth and access parameters in the network implementation plan**—In the network implementation plan, you can configure the bandwidth allowed through a device and specify the VLANs that contain the device ports. This information is used by Service Orchestration while allocating resources for provisioning a service.

[See [Add a Network Implementation Plan](#).]

- **Monitor network trust**—Paragon Automation provides a dashboard that enables you to monitor network health-related data in real time. On the Trust Tab (**Observability > Health > Health Dashboard > Trust**) of the dashboard, you can view the overall trust score and monitor, in real time, vulnerabilities that affect targets in the network. You can use this information to perform corrective actions when necessary.

The Trust tab consists of:

- **Trust Pane**—View the trust score of the targets, trust plan applied to the network, overall trust trend, and a graph of the average trust score for the past 30 days.
- **Vulnerabilities Accordion**—View the percentage of healthy devices and the total number of unhealthy devices based on device vulnerabilities, and a graph that displays the average health of all devices. You can also view KPIs such as proactive bug notifications (PBNs) and advisories that affect overall network health. Click **View Details** to view the percentage of healthy devices and KPIs. You can also view the total number of unhealthy devices grouped by OS version and device model.

[See [About the Trust Tab.](#)]

- **Manage allocation of devices and interfaces for service provisioning**—Placement is the process of allocating network resources for provisioning services. Use the **Update Placements** button to automatically assign all possible placement options available for a customer to provision L3VPN and EVPN services. After placement options are assigned, you can select and allocate network resources such as PE devices and interfaces, VLANs, and so on from the available options to provision services for the customer.

[See [Manage Placement Configurations for Service Instances.](#)]

- **Use IPv6 for provisioning infrastructure service**—Paragon Automation supports the use of IPv6 addresses to provision infrastructure service. BGP, OSPF, loopback and regular interfaces support IPv6 addresses for infrastructure service provisioning. Paragon Automation supported only IPv4 addresses in earlier releases.

[See [Create a Layer 3 Resource Pool.](#)]

## Known Issues

### IN THIS SECTION

- [Device Life-Cycle Management | 13](#)
- [Observability | 16](#)
- [Service Orchestration | 21](#)
- [Active Assurance | 26](#)
- [Trust | 27](#)
- [Administration | 27](#)
- [Installation and Upgrade | 27](#)

This section lists the known issues in Juniper Paragon Automation.

# Device Life-Cycle Management

- If the device onboarding fails, the device onboarding status is displayed as *Status is not available* in the Devices section of the Network Implementation Plan page (**Inventory > Device Onboarding > Network Implementation Plan**).

Workaround: For such devices, initiate the outbound SSH connection on the router so that the onboarding workflow restarts.

- In this release, single node failure and restart might cause inconsistencies in device inventory.

Workaround: None.

- Sometimes, the onboarding workflow might restart for a device that is already onboarded. This is harmless, as the workflow will observe that the node is already onboarded, report the observation, and exit.

Workaround: None.

- Paragon Automation triggers the configuration templates included in a device profiles and interface profile only during the initial onboarding of the device. You cannot use the configuration templates included in the device profiles and interface profiles to apply additional configuration on a device after the device is onboarded.

Workaround: If you need to apply additional configuration on a device after the device is onboarded, you need to manually apply the configuration.

- If you have enabled the Trust option in the device profile, the onboarding workflow may fail with the following error:

Onboarding workflow failed reason: task\_failure, Trust score computation failed.

Workaround: Retry the onboarding process after a few minutes.

- Airflow scheduler pod may crash when multiple onboarding DAGs are triggered simultaneously.

By default, Airflow allocates 10 GB of disk space in the **PVC mount** directory (`/opt/airflow/mount/`). When multiple onboarding DAGs are triggered simultaneously, excessive logs are generated thereby filling up the disk space and potentially causing the Airflow scheduler pods to crash.

Workaround: To avoid this issue, we recommend that you limit onboarding to 100 devices over a 12-hour period. In case of disk space issue, use the following procedure to clean up the logs:

1. Log in to a controller node and enter in to airflow worker pods.

```
root@masternode1:~# kubectl get pods -n airflow | grep airflow-worker
airflow-worker-86d88445bd-9z9jn      1/1      Running    0           4d4h
airflow-worker-86d88445bd-wb6q8     1/1      Running    0           39m
airflow-worker-86d88445bd-xfg7m     1/1      Running    0           4d4h

root@masternode1:~# kubectl exec -it airflow-worker-86d88445bd-9z9jn -n airflow -- bash
Defaulted container "worker" out of: worker, init-data-models (init)
```

2. Delete all log files from `/opt/airflow/mount/logs/`.

```
root@airflow-worker-86d88445bd-9z9jn:/opt/airflow# rm -rf /opt/airflow/mount/logs/*
root@airflow-worker-86d88445bd-9z9jn:/opt/airflow# exit
```

3. Restart the airflow-scheduler pods.

```
root@masternode1:~# kubectl get pods -n airflow | grep airflow-scheduler
NAME                                READY   STATUS    RESTARTS      AGE
airflow-scheduler-8474cf496-kvdtb   2/2     Running   329 (17h ago)  4d4h
airflow-scheduler-8474cf496-mttn7   2/2     Running   323 (17h ago)  4d4h

root@masternode1:~# kubectl delete pod airflow-scheduler-8474cf496-kvdtb -n airflow
pod "airflow-scheduler-8474cf496-kvdtb" deleted
root@masternode1:~# kubectl delete pod airflow-scheduler-8474cf496-mttn7 -n airflow
pod "airflow-scheduler-8474cf496-mttn7" deleted
```

- If you try to adopt Cisco devices using the GUI or try to register Cisco devices in bulk using the REST APIs then the system does not trigger a connection to the devices. Therefore, the device status is shown as disconnected.

Workaround: You can add Cisco devices using the Register Device REST API.

- You cannot release a Cisco device using the **Release Router** option. This issue occurs if you have added the Cisco device to Paragon Automation from the Inventory page (**Inventory > Devices > Network Inventory**) and you have used non-alphanumeric characters for the MAC address.

Workaround: None.

- Onboarding an ACX7024 device fails with the following error:

task\_failure, Failed to launch Software update. Timed out or waiting for the launch message. Try again later.



Workaround: Restart the onboarding process using the service orchestration cMGD CLI. Perform the following steps:

1. Log into the primary node using SSH.
2. Exit out of the default paragon CLI to the Linux root shell.
3. Log in to the service orchestration CMGD CLI.

```
kubectl exec -it -n foghorn deployment/cmgd -- bash
cli
```

4. Get the organization UUID from the GUI. To find your organization's UUID, go to **Administration > Settings** in Paragon Automation and then copy the organization's UUID.
5. Configure the organization UUID in the cMGD CLI:

```
configure
set foghorn:core org-id org-uuid
commit and-quit
```

6. Retrieve the service order.

```
show service order status customer-id_instance-id
request service project add /data-models/projects/network-resource.tgz
request service project add /data-models/projects/onboard.tgz
request service order sync customer-id_instance-id
```

7. Reset the node state.

```
configure
delete foghorn:customers customer customer-id services instance-id infrastructure
infrastructure-ntw network-nodes network-node <node-name> onboard
set foghorn:customers customer customer-id services instance-id infrastructure
infrastructure-ntw network-nodes network-node <node-name> node-type paper-node
commit and-quit
request service order load merge customer-id_instance-id
```

8. Log in to the router.

9. Restart the onboarding by restarting the connection.

```
configure
  deactivate system services outbound-ssh
  commit
  activate system services outbound-ssh
  commit and-quit
```

## Observability

- The Hardware accordion does not display the following information for the listed devices:
  - Chassis temperature (chassis-temperature) for MX204, MX240, MX304, MX10004, MX10008, and MX10016 devices.
  - Charts related to the fan speed (rpm-percent) for MX480, MX960, MX10004, MX10008, and MX10016 devices.
  - Power supply module temperature (psm-temperature) for MX204, MX480, and MX960 devices.
  - Line card charts for some ACX Series and MX Series devices as the flexible PIC concentrator (FPC) fields are not supported on these devices. See [Table 2 on page 16](#) for more information.

**Table 2: Line Card Charts Support**

| Device Family | Device Series                                                 | FPC Fields Not Supported                                            |
|---------------|---------------------------------------------------------------|---------------------------------------------------------------------|
| ACX Series    | ACX7100-32C, ACX7100-48L, ACX7024, ACX7024X, ACX7509, ACX7348 | fpc-temperature, fpc-cpu-utilization, fpc-buffer-memory-utilization |
| MX Series     | MX204, MX240, MX304, MX480, MX960, MX10004, MX10008, MX10016  | fpc-temperature, fpc-cpu-utilization                                |

- On the Interfaces accordion, forward error correction (FEC) corrected errors and FEC uncorrected errors charts are available only on interfaces that support speeds equal to or greater than 100-Gbps.

- After you apply a new configuration for a device, the Active Configuration for *Device-Name* page (**Observability > Troubleshoot Device > Device-Name > Configuration accordion > View active config link**) does not display the latest configuration immediately. It takes several minutes for the latest changes to be reflected on the Active Configuration for *Device-Name* page.

Workaround: You can verify whether the new configurations are applied to the device by logging in to the device using CLI.

- The graphs related to cyclic redundancy check (CRC) errors display *No Results Found* as the CRC errors-related data is not streamed from the devices for the management ports.

Workaround: None.

- If a device is discovered through a BGP-LS peering session even before you onboard the device, then duplicate LSPs are created when a PCEP session is established with the device. In some rare cases, the duplicate LSPs may remain.

Workaround: If you see duplicate LSPs, restart the EdgeAdapter pod.

- For PTX10001, PRX10004, and PTX10016 devices, the PSM Temperature graph on the Hardware Details for *Device-Name* page (**Observability > Troubleshoot Devices > Device-Name > Hardware accordion > PSUs**) does not display any data.

Workaround: None.

- After the primary node is switched off, the OC-term and GNMI state on the Remote Management accordion (**Observability > Troubleshoot Devices > Device-Name**) are displayed as disconnected.

Workaround: You can do one of the following:

- Offboard and onboard the devices, or
- Restart the OC-term pod.
- Sometimes, the graph on the Output Traffic Details for *Device-Name* page (**Observability > Health > Troubleshoot Devices > Device-Name > Overview > Interfaces (accordion) > Output Traffic data link**) displays a sudden spike in data. This issue occurs because some devices might erroneously send data more than once with the same time stamp.

Workaround: None.

- Sometimes, data points on the graphs on the Input/Output Traffic Details for Device-Name pages (**Observability > Health > Troubleshoot Devices > Device-Name > Overview > Interfaces (accordion) > Input/Output Traffic data link**) are displayed as critical even though the traffic is below the threshold level.

When the LACP state of a child interface is down, the aggregated interface speed also comes down based on how many child interfaces are active at the point of data collection. However, when the

LACP state of the child interface is up, the aggregated interface speed also increases. Because the child interface LACP state is toggling, the aggregated interface speed also toggles and alerts are raised as the input/output traffic utilization is beyond the threshold with the reduced speed. But in the graph, since the latest thresholds are plotted, alerts might be raised even though the traffic is below the threshold.

Workaround: You can check any flaps on aggregated or child interface on the **Observability > Health > Troubleshoot Devices > *Device-Name* > Overview > Interfaces (accordion) > Interfaces > Link Flap** performance graph. In addition, you can also execute the `run show LACP interfaces` command to check the interface flap data and correlate these issues with the child interfaces LACP states being flapped.

- The number of unhealthy devices listed on the Troubleshoot Devices and Health Dashboard pages (**Observability > Health**) do not match.

Workaround: None.

- You cannot delete unwanted nodes and links from the Paragon Automation GUI.

Workaround: Use the following REST APIs to delete nodes and links:

- REST API to delete a link:

[DELETE] `https://{server_ip}/topology/api/v1/orgs/{org_id}/{topo_id}/links/{link_id}`



**NOTE:** You can follow the steps described ["here" on page 19](#) to get the actual URL.

For examples,

- URL: 'https://10.56.3.16/topology/api/v1/orgs/f9e9235b-37f1-43e7-9153-e88350ed1e15/10/links/15'
- Curl:

```
curl --location --request DELETE 'https://10.56.3.16:443/topology/api/v1/orgs/
f9e9235b-37f1-43e7-9153-e88350ed1e15/10/links/15' \
--header 'Content-Type: application/json' \
--header 'Authorization: Basic dGVzdDFAdGVzdC5jb206RW1iZTFtcGxz'
```

- REST API to delete a node:

[DELETE] `https://{Server_IP}/topology/api/v1/orgs/{Org_ID}/{Topo_ID}/nodes/{Node_ID}`



**NOTE:** You can follow the steps described "[here](#)" on page 19 to get the actual URL.

For examples,

- URL: 'https://10.56.3.16/topology/api/v1/orgs/f9e9235b-37f1-43e7-9153-e88350ed1e15/10/nodes/1'
- Curl:

```
curl --location --request DELETE 'https://10.56.3.16:443/topology/api/v1/orgs/
f9e9235b-37f1-43e7-9153-e88350ed1e15/10/nodes/11' \
--header 'Content-Type: application/json' \
--header 'Authorization: Basic dGVzdDFAdGVzdC5jb206RW1iZTFtcGxz' \
```

Use the following procedure to get the actual URL that you use in CURL for deleting a link or a node:

1. Navigate to the Topology page (**Observability > Topology**).
  2. Open the developer tool in the browser by using the **CTRL + Shift + I** buttons in the keyboard.
  3. In the developers tool, select **Network** and select the **XHR** filter option.
  4. Identify the link index number or node number. To identify the link index number or the node number:
    - a. On the Topology page of the Paragon Automation GUI, double click the link or the node that you want to delete.  
  
The Link *Link-Name* page or the Node *Node-Name* page appears.
    - b. Navigate to the Details tab and note the link index number or the node number that is displayed.
  5. In the developers tool, select and click the row based on the link index number or the node number that is related to the link or the node that you want to delete.
  6. Copy the URL that you need to use to delete the link or node in CURL.
- The values for Input Utilization and Output Utilization are represented in terms of percentage of interface utilization instead of Mbps.

This issue is seen in Physical Interfaces Details for *Service-Instance-Name* tab (**Orchestration > Instances > Service Instances > *Service-Instance-Name* hyperlink > *Service-Instance-Name* Details > Passive Assurance**) and Input Traffic Details for *Device-Name* page (**Observability >**

Health > Troubleshoot Devices > *Device-Name* > Overview > Interfaces (accordion) > Input Traffic data-link).

Workaround: To get the actual traffic rate value, you should multiply the values displayed by interface speed.

- Not all optics modules support all the optics-related KPIs. See [Table 3 on page 20](#) for more information.

Workaround: None.

**Table 3: KPIs Supported for Optics Modules**

| Module            | Rx Loss of Signal KPI | Tx Loss of Signal KPI | Laser Disabled KPI |
|-------------------|-----------------------|-----------------------|--------------------|
| SFP optics        | No                    | No                    | No                 |
| CFP optics        | Yes                   | No                    | No                 |
| CFP_LH_ACO optics | Yes                   | No                    | No                 |
| QSFP optics       | Yes                   | Yes                   | Yes                |
| CXP optics        | Yes                   | Yes                   | No                 |
| XFP optics        | No                    | No                    | No                 |

- The Connectivity accordion (**Observability > Troubleshooting Devices > Device-Name**) shows no data although connectivity tests are properly run and passed.

Workaround: For the connectivity accordion to display data, use the following REST API:

```
`/active-assurance/api/v2/orgs/${org_id}/streams?
filter=measurement.metadata.tags.__sys__test_execution_id:'${t.id}'`
```

# Service Orchestration

- The "vpn\_svc\_type" service type is displayed as "pbb-evpn" instead of "evpn-mpls" on the Paragon Automation GUI and through the REST API.

Workaround: None.

- The following limitations are seen when you use the service orchestration cMGD CLI to modify the placement-interface information of an L3VPN service:
  - The initial placement-interface options that were populated when the service order was created are not displayed.
  - You can select the interface for the site access from all the interfaces present on the CE or PE device.
  - When you modify the PE topology and the available ports in the topology, you must:
    1. Delete the existing placement-interface and placement-options from the site network access by using either REST API or the service orchestration cMGD CLI.
    2. Execute the `request service order modify` command to regenerate the service order with the modified values for the placement-options.
- Sometimes, the apply insights configuration (**appy\_insights-config**) fails if you try to provision a service without properly deleting a previously provisioned service or a device.

For example, if you release the router without off-boarding or deleting a service, then the apply insights configuration fails when the same service or device is used in another organization.

**Workaround:**

- If there are stale services and devices, run the following REST APIs from the cMGD container of the **foghorn** namespace to delete stale services and devices, and rerun the workflow:
  - `curl --request DELETE <http://config-server.healthbot:9000/api/v2/config/services/device-group/<device-group> name>/`
  - `curl --request DELETE <http://config-server.healthbot:9000/api/v2/config/device-group/<device-group> name>/`
  - `curl --request POST http://config-server.healthbot:9000/api/v2/config/configuration/`
- If there are stale network-groups, run the following REST APIs from the cMGD container of the **foghorn** namespace to delete the stale network-groups, and rerun the workflow:

- `curl --request DELETE <http://config-server.healthbot:9000/api/v2/config/services/device-group/<network-group> name>/`
- `curl --request DELETE <http://config-server.healthbot:9000/api/v2/config/device-group/<network-group> name>/`
- `curl --request POST http://config-server.healthbot:9000/api/v2/config/configuration/`
- While configuring an EVPN service order, the GUI does not throw a validation error even if you specify a value that is equal to 1 Tbps for CBS and CIR fields.

Workaround: Based on your topology, ensure that you specify the right values for CBS and CIR fields.

- The EVPN service order creation fails if you try to create an EVPN service order by importing an existing JavaScript Object Notation (JSON) file.

Workaround: If you are using a JSON file, ensure that you clear the placement section before you publish the service order.

- For some devices such as ACX7204, if you configure VLANs on unused ports, the following error occurs:

VLAN must be specified on tagged interfaces.

Workaround: This issue is caused by the default factory configuration on the port. Delete the default factory configuration on the ports that you plan to use.

- For an MX 240 device, the OSPF-related data is not populated on the Passive Assurance tab (**Orchestration > Instances > *Service-Order-Name* Details**).

Workaround: Configure OSPF on the customer edge (CE) device.

- Although multiple VLAN IDs are available in the topology resources, the Placement section of the EVPN service order lists only one VLAN ID in the drop down.

Workaround: To fix this issue:

1. Edit the EVPN service order to add new VLAN IDs. You can add the VLAN IDs under the Tagged Interface section.
  2. Clear the Placement section by deselecting the device name.
  3. Save and publish the service order.
- While modifying a service order, you cannot clear the existing placements.

Workaround: If publishing the service order fails due to existing placements, you can export the failed service order to JSON format and then create a new service order or modify an existing service



order by importing this JSON file. During the importing process, delete the placements and then publish the service order.

- While creating or modifying an EVPN service order, you cannot configure multiple VLAN IDs on the Aggregated Ethernet (AE) interface. The EVPN considers the AE port as a single resource and therefore an AE interface cannot be reused across service instances even when the VLAN IDs on the AE IFL differ.

Workaround: None.

- Scheduling provisioning of service orders is a Beta feature in Release 2.1.0. Except in fresh installations, scheduling may not work consistently.

Workaround: None.

- VLAN drop down under Placement section doesn't display all the available VLANs as per the topology service order and it shows only the selected VLAN.

Workaround: None.

- While modifying a resource instance, if you update VLAN with a value higher than the current specified value then the Modify Resource Instance operation fails.

Workaround: None.

- The service order fails with the error message, Invalid XML document, namespace is missing.

Workaround: On the device with the failed configuration, you should turn off system services netconf rfc-compliant and system services netconf notification knobs.

- During device onboarding, pings from the device to Microsoft Azure and Google Cloud Platform endpoints fail.

Workaround: Instead of Microsoft Azure and Google Cloud Platform, use Amazon Web Services (AWS) as the endpoint.

- While creating an EVPN service order, if the MAC Address Limit that you have specified is out of the defined range then the service order fails.

Workaround: Specify a value that is within the defined range, and then republish the service order.

- While creating or modifying an EVPN service order, the MAC Address Limit configuration is ignored if you specify the action to be taken as Drop when the upper limit for customer MAC addresses exceed.

Workaround: None.

- Since you can manage resources from within network implementation plans, a blank topology instance (*topo*) is auto-created on the Resources Instances page (**Orchestration > Service > Resource Instances**). This topology instance is a read-only instance and is owned by the network operator.

Workaround: None.

- If you have referenced a device in a service order and if you try to release the router from the organization, then you cannot delete the service.

Workaround: We recommend that you deprovision the service and offboard the device before you release the router from the Network Inventory page.

- When you click the **Refresh** icon on the *Service-Instance-Name* Details page (**Orchestration > Instances > Service-Instance-Name**), you may not see the latest events in the Relevant Events section.

Workaround: To view the latest events, instead of using the refresh icon go to the Service Instance page (**Orchestration > Instances**) and select the service instance for which you need to see the latest events.

- After you upgrade Paragon Automation from Release 2.1.0 to Release 2.2.0, L2VPN and L3VPN accordions (**Orchestration > Instances > Service Instances > Service-Instance-Name hyperlink > Service-Instance-Name Details > Passive Assurance tab**) show no data for service instances that were provisioned in Release 2.1.0. This issue does not occur if you install Paragon Automation afresh.

Workaround: You need to recreate the VPN service after you upgrade to Juniper Paragon Automation Release 2.2.0.

- On the Customer Inventory page (**Orchestration > Customers**), when you click the **View Instances** hyperlink to view the service instances provisioned for the customer, a filter for the Customer field is automatically applied on the Service Instances page. Since filtering is not supported for the customer field, you may not see any data.

Workaround: Clear the filter by clicking the **X** icon in the advanced filtering section. You can then view all the service instances provisioned for the customer.

- While creating an L3VPN service instance if you click **Reset Placements**, the resources are cleared from the UI. However, these resources are not released from the back end.

Workaround: To clear any reserved resources, you can either provision or deprovision the service.

- When you click **Update Placements** while modifying services, a confirmation message appears that the placements are successfully updated even though the REST API returns an error.

Workaround: None.

- You must configure access diversity while creating an EVPN service order with all-active or single-active redundancy modes. However, the Access Diversity section is not populated in the UI.

Workaround: None.

- The order state of a network implementation plan does not reflect the onboarding status of individual devices in the plan. The order status may show as a success even if the device onboarding has failed.

Workaround: We recommend that you view the details of a network implementation plan so that you can see the onboarding status of individual devices.

To view the details of a network implementation plan, you can select the network implementation plan on the Network Implementation Plan page (**Inventory > Device Onboarding**) and click **More > Detail**. Alternatively, hover over the plan name and click the Details icon that appears.

- While modifying an existing L3VPN instance, if you try to remove a device that is already a part of the network implementation plan then the modify workflow fails.

Workaround: You must delete all *paa-\** groups and *apply-groups* from all the devices used in the instance.

- After you upgrade Paragon Automation from Release 2.1.0 to Release 2.2.0, the *vpn* resource instance does not appear on the Resource Instances page (**Orchestration > Service > Resource Instances**). This issue occurs because the *vpn* service design has been renamed to *vpn-resources* in Release 2.2.0.

Workaround: We recommend that you delete the *vpn* resource instance, and create a new resource instance, *vpn-resources*. Ensure that the *vpn-resources* resource instance has the same data as that of the *vpn* resource instance.

- While provisioning an EVPN service order, if you mark an access interface as untagged, the placement may still pick up a tagged interface from interface resources defined in the topology file.

For example, if et-0/0/2 is defined as untagged and et-0/0/3 is defined as a tagged interface in the topology resources, then the service order sets the access interface as untagged and the automatic placement may still pick up the tagged et-0/0/3 interface.

Workaround: Go to the placement section and manually choose the desired untagged interface from the drop down.

- If you modify an L3VPN service to update the routing protocol then the changes are not reflected in the L3VPN accordion (**Orchestration > Instances > Service Instances > *Service-Instance-Name* hyperlink > Passive Assurance**).

Workaround: Instead of modifying the existing service, create a new service with the new routing protocol.

- Scheduling provisioning of service orders does not work if you upgrade Paragon Automation to Release 2.2.0.

Workaround: None.

- The Order History tab on the *L3VPN-Name* Details page (**Orchestration > Instances > *Service-Instance-Name*** hyperlink) lists all the order history even if the service instance is deprovisioned and provisioned again.

Workaround: None.

- We recommend that you do not enable LDP for IPv6 for the L2 circuit.

Workaround: None.

- Sometimes, the Interface Status column on the Passive Assurance tab of the *L3VPN-Name* Details page (**Orchestration > Instances > *Service-Instance-Name*** hyperlink) may not display data. Occasionally, you may not see any data on the Interface Status graph.

Workaround: None.

## Active Assurance

- On the Tests page (**Observability > Active Assurance**), the Test summary that is displayed on the info card is not based on the time range that you have selected. Instead, the Test summary is based on the number of Tests listed on a specific page.

Workaround: None.

- Sometimes you may notice that the previously-available Test Agents and previously-provisioned Monitors are not shown in the GUI. In addition, the performance tests are timing out and the measurements are stopped.

Workaround: None.

- You cannot delete a Test from the Test-Name page (**Observability > Active Assurance > Test > Test-Name > *Test-Name* > More > Delete**). The following error is displayed and the Test that you tried to delete is retained:

Failed to delete Test

Workaround: None.

- The status of a Test Agent is shown as offline after the device's Routing Engine switches over from the primary Routing Engine to the backup Routing Engine, or vice versa.

Workaround: Reinstall Test Agent after the Routing Engine switchover.

- The streams are not generated when you create a Test with a DNS plug-in and the following event is raised:

Could not get nameserver from resolv.conf

This issue occurs when the Test is associated with a Test Agent that runs on a Juniper Networks router with Junos OS EVO installed, and you don't specify the Name Server field while configuring a Test.

Workaround: Ensure that you specify a value for the Name Server field while configuring a Test.

## Trust

There are no known issues in this release.

## Administration

There are no known issues in this release.

## Installation and Upgrade

- The backup and restore functionality has the following caveats:
  - You cannot restore data backed up from a setup with a different release of Paragon Automation to a setup with the current release.
  - Paragon Automation backs up only application configurations such as devices, sites, service orders, and so on. Since a backup does not store the certificates and infrastructure services configurations, that information must be kept unchanged during restoration.
  - Resources allocated to the network won't be preserved after a restore and you must ensure that you release the allocated resources during the window between taking a backup and performing a restore.

Workaround: None.

- If the PCE Server VIP address is not configured, kube-proxy is set to a random port.

Workaround: Configure the PCE Server VIP address.

- If a node in the cluster is not operational, the status of the vector pod from the node that is not operational is displayed as *Running*, even though the node status is reported as *Not Ready*. This is due to an existing Kubernetes issue. See <https://github.com/kubernetes/kubernetes/issues/117769>.

Workaround: You can do one of the following:

- Monitor the metric, *kube\_daemonset\_status\_number\_ready*. When the value for this metric drops to three, you can manually check from which vector the data is missing.
- Set a query and an alert for the *kube\_daemonset\_status\_number\_ready* metric in Grafana.
- You might encounter RKE2-related issues if you change the hostname after you set up a cluster.

We recommend that you do not change the hostname after a cluster is set up.

Workaround: None.

- When the worker node is down, there might be issues if you create an organization or onboard a device.

Workaround: Do not create an organization or onboard a device when a worker node is down. You must wait until the cluster recovers and then create an organization or onboard a device. Recovered state is when all the pods are either in *Running* or *Pending* state and are not in any intermediate states like *Terminating*, *CrashloopbackOff*, and so on.

- If you have powered off one of the primary nodes, you may not be able to log in to the Juniper Paragon Automation GUI.

Workaround: Restart papi-ws using the following Paragon Shell CLI command:

```
request paragon cluster pods reset service papi-ws namespace papi operation restart
```

- Upgrading from Release 2.1.0 to Release 2.2.0 on VMs configured with minimum required specifications might fail due to a failed airflow-worker instance. Verify the reason for failure.

```
root@pa1:~# kubectl get pods -A | grep -v Running
```

| NAMESPACE               | NAME                          | READY   | STATUS |
|-------------------------|-------------------------------|---------|--------|
| airflow                 | airflow-                      |         |        |
| worker-6cc5484f95-78cfv | 0/1                           | Pending | 0 30m  |
| healthbot               | insight-db-create-job-        |         |        |
| cgvzz                   | 0/2 Completed                 | 0       | 2d11h  |
| kube-system             | helm-install-rke2-calico-crd- |         |        |
| zs8pg                   | 0/1 Completed                 | 0       | 2d11h  |

```

kube-system                                helm-install-rke2-calico-
qzbnw                                     0/1    Completed    1          2d11h

<output snipped>

rook-ceph                                rook-ceph-osd-prepare-q-eop-dell-58-vm4-
cds6v                                     0/1    Completed    0          6h38m
root@pa1:~# kubectl describe pods airflow-worker-6cc5484f95-78cfv -n airflow
Name:          airflow-worker-6cc5484f95-78cfv
Namespace:     airflow
Priority:       0

<output snipped>

Tolerations:      node.kubernetes.io/not-ready:NoExecute op=Exists for 30s
                  node.kubernetes.io/unreachable:NoExecute op=Exists for 30s
Events:
  Type    Reason             Age          From          Message
  ----    -
  Warning FailedScheduling   21m          default-scheduler  0/4 nodes are available:
  4 Insufficient cpu. preemption: 0/4 nodes are available: 4 No preemption victims found for
  incoming pod.
  Warning FailedScheduling   5m10s (x4 over 20m) default-scheduler  0/4 nodes are available:
  4 Insufficient cpu. preemption: 0/4 nodes are available: 4 No preemption victims found for
  incoming pod.
root@pa1:~#

```

Workaround: Perform the following steps:

1. Lower the following CPU resource requests from the Linux root shell.

```

# kubectl patch deployment -n mems mems --type "json" -p '[{"op":"add","path":"/spec/template/spec/containers/0/resources/requests/cpu","value":"0.5m"}]'# kubectl patch deployment -n tagify tagify --type
"json" -p '[{"op":"add","path":"/spec/template/spec/containers/0/resources/requests/cpu","value":"0.5m"}]'

```

2. Rerun upgrade.

- On rare occasions, e-mails might not be sent out due to Kafka connection issues. The mail service logs display the following error:

```
"message":"dial tcp 10.x.x.43:9092: connect: connection refused"
```

This issue can occur anytime, but more commonly when the cluster has been upgraded or when a cluster node has been rebooted.

Workaround: Restart the mail server from the Linux root shell.

```
kubect1 delete pods -n common mailservice-xxxxxxxxxx-xxxxx
```

## Resolved Issues

This section lists the issues resolved in Juniper Paragon Automation Release 2.2.0:

- When you use the **Release Router** option to release a device from being managed by Paragon Automation, the device might not be released as the service orchestration engine might still be referencing the device that you want to release.
- The onboarding of a device fails if you use NETCONF EDIT or NETCONF RPC configuration formats in the configuration templates.
- A warning message is not displayed when you try to delete a configuration template that is used in the network implementation plan.
- For PTX10001, PRX10004, and PTX10016 devices, the Linecards graph on Hardware details for *Device-Name* (**Observability > Troubleshoot Devices > Device-Name > Hardware accordion**) page does not display any data.

Workaround: None.

- For PTX Series, MX Series, and ACX Series devices, the RSVP TE Global Errors graph on the RSVP Routing Details for *Device-Name* (**Observability > Troubleshoot Devices > Device-Name > Routing and MPLS accordion**) page does not display any data.
- If you modify an existing Monitor and then restart the Monitor, the events that are raised before modifying the Monitor are not cleared.
- The latest audit log messages may not be displayed on the Audit Logs page.
- The history of service orders that are generated for a service instance is saved in the Order History Tab for auditing purposes. However, when you delete the service instance, the service order history gets deleted.
- Sometimes, the publishing of a service order fails due to existing placements.
- When you edit a topology resource instance, the POP page may not list the latest sites or nodes.
- The publishing of an EVPN service order fails if you modify an existing EVPN service order to add a new site. This issue occurs only when you modify an existing EVPN using the GUI.
- The **VPN Node Id** field on the VPN Node page (**Service Orchestration > Instances > Add L2 Circuit**) field may not list all the onboarded devices.



- While creating an EVPN site, if the value that you have specified for the **Minimum number of links** field is greater than the number of Member Links, then the EVPN service order fails.
- If you try to modify an existing resource instance that does not have a device or a site, then the Inventory section of the Modify *Resource-Instance-Name* page does not load.
- Even though the EVPN service order is successfully created and the configurations are pushed to the devices, the following error message is displayed and the communication between the customer Edge (CE) devices fails:

Status: Configuration Error

This issue occurs in a multi-homing scenario (multiple provider edge (PE) devices are connected to the same customer edge device) with *single-active* or *all-active redundancy* mode.

- All access circuits must have the same VLAN configured, failing which the service may not function as desired.
- If there are multiple node failures, the OpenSearch database may fail to start and rejoin the cluster.
- If the PCE Server VIP address is not configured, kube-proxy is set to a random port.
- There is no synchronization between accessing and configuring devices. A workflow might fail if a device is configured by more than one service order.
- If you modify an L3VPN service, the historical data for Monitors related to the modified L3VPN service are deleted.
- The REST API, **api-aggregator**, does not capture alerts that are related to Tests on the Connectivity accordion (**Observability > Troubleshooting Devices > Device-Name**).