Juniper Networks | Engineering Simplicity

Juniper Apstra Device Replacement

Published
2024-03-07

RELEASE

Table of Contents

Juniper Apstra Device Replacement Guide

IN THIS SECTION

An essential operation in a working data center network is the need to replace a device that failed or needs to be re-allocated or reused for other purposes. This document describes how to replace a Juniper Apstra managed device using the Apstra GUI and Apstra Terraform Provider.

Replace a Managed Device Using the Apstra GUI

IN THIS SECTION

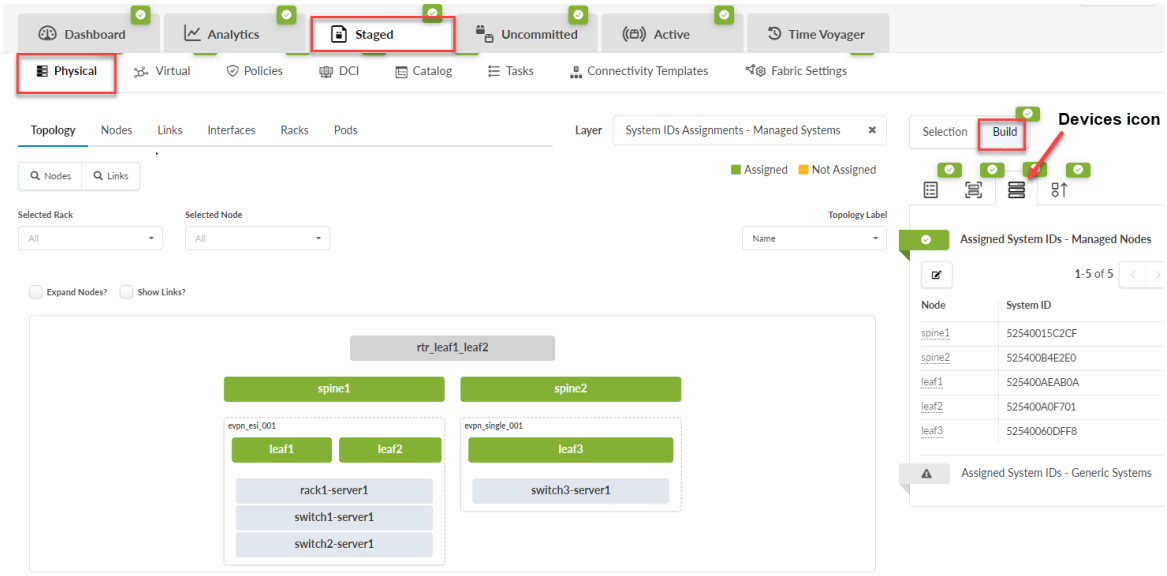
Replacing a managed device in the Apstra GUI is a straightforward set of tasks. The managed devices are abstracted from their role in the fabric using Logical Devices and Interface Maps and are assigned to a blueprint.

NOTE: In these procedures, we are replacing the leaf2 device in the GUI. The replacement device (leaf2) is exactly the same physical device (make, model and port configuration) as the device we are replacing.

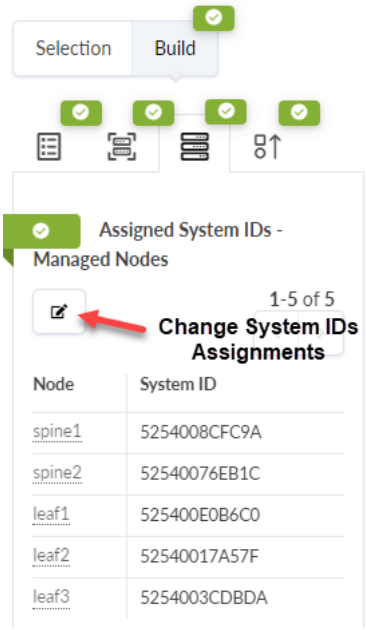
In this topic, we'll show you how to replace a device using the following methods:

Delete a Leaf with a Still Connected Device

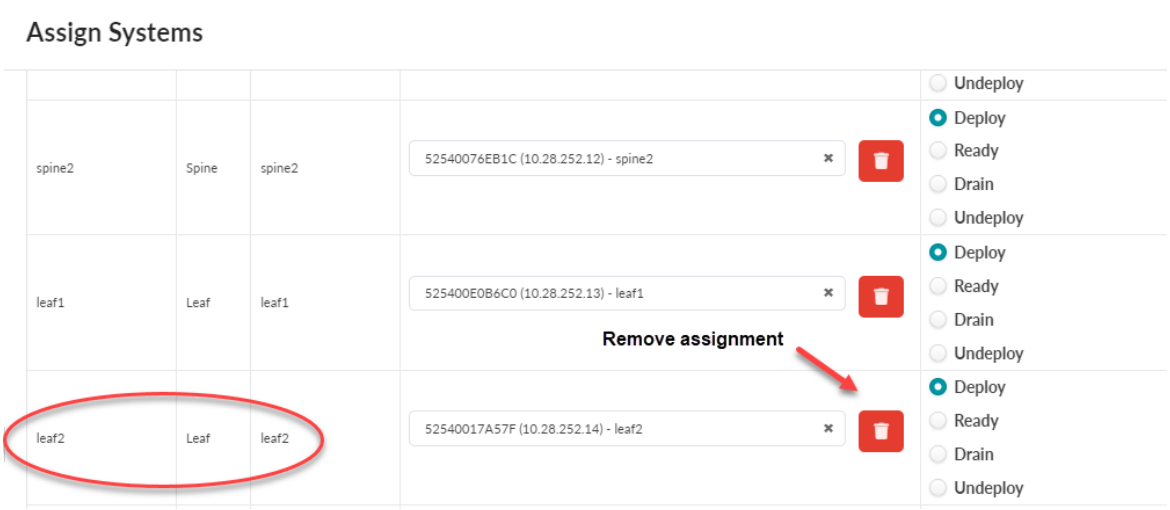
1. From the left navigation menu in the Apstra GUI, navigate to **Blueprints > Blueprint ID > Staged > Physical > Build**, then click the **Devices** icon.



2. Click **Assigned System IDs - Managed Nodes** , then click the **Change System IDs Assignments** icon.

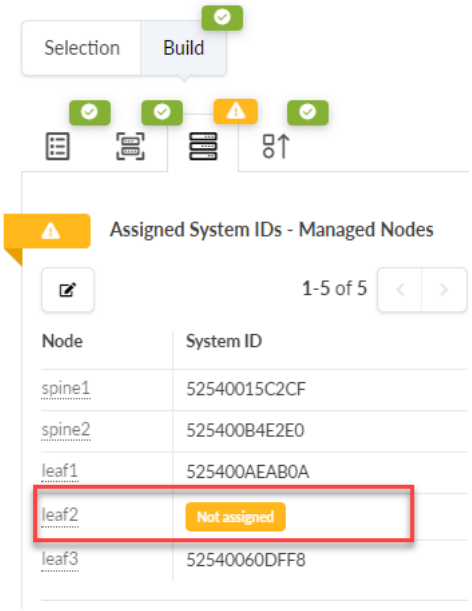


3. Click the trash icon, then click **Update Assignments** to unassign leaf2.

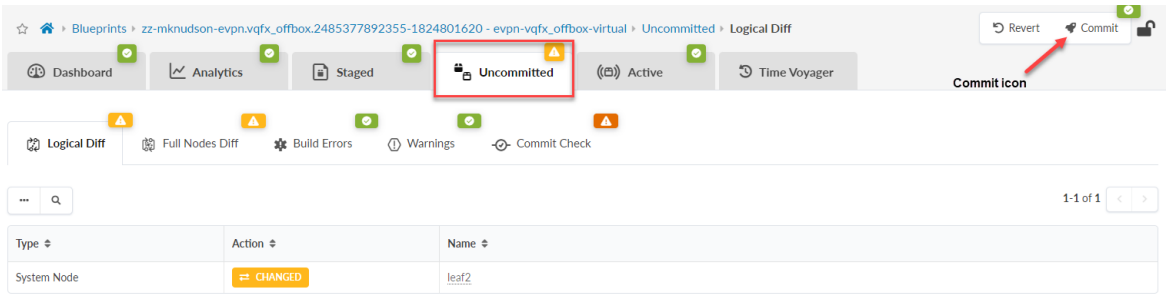


Here we can see that leaf2 in no longer assigned.

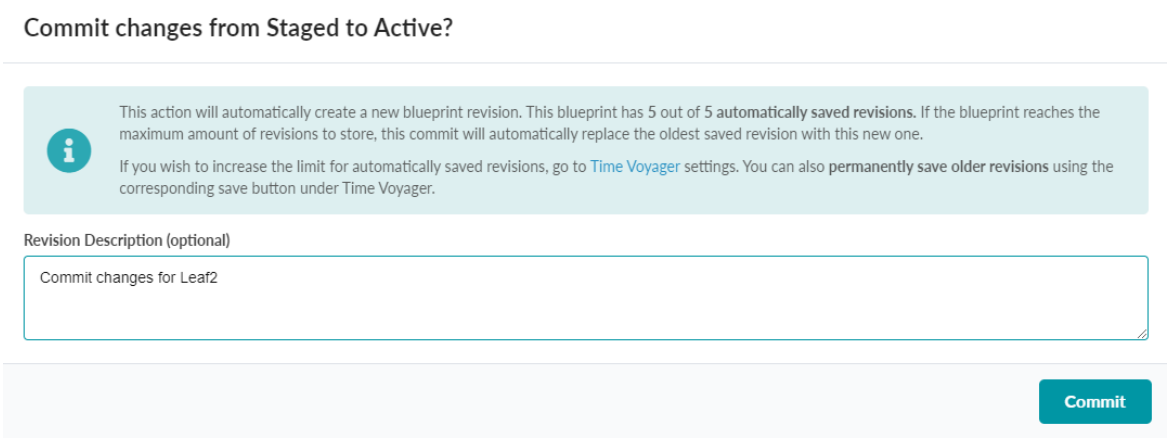
4.



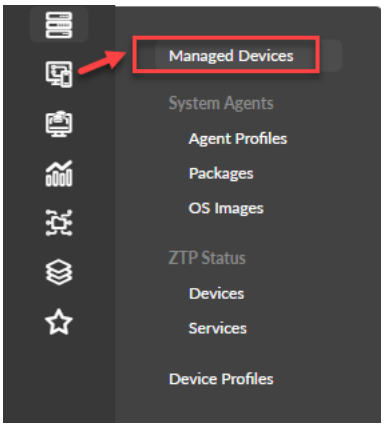
5. Navigate to the **Uncommitted** tab and click the **Commit** rocket icon.



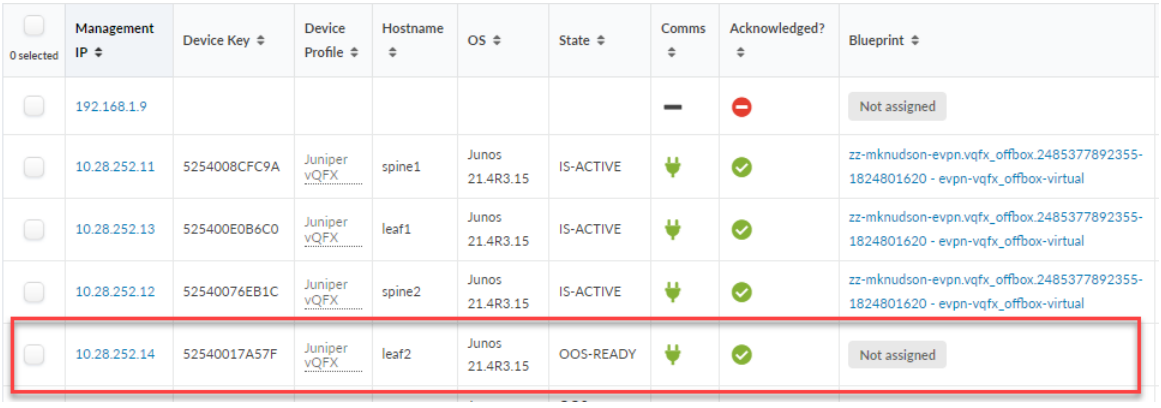
6. Add a comment if desired, and click **Commit**.



7. When the commit is complete, navigate to the **Devices > Managed Devices** to return to the Managed Devices table view.

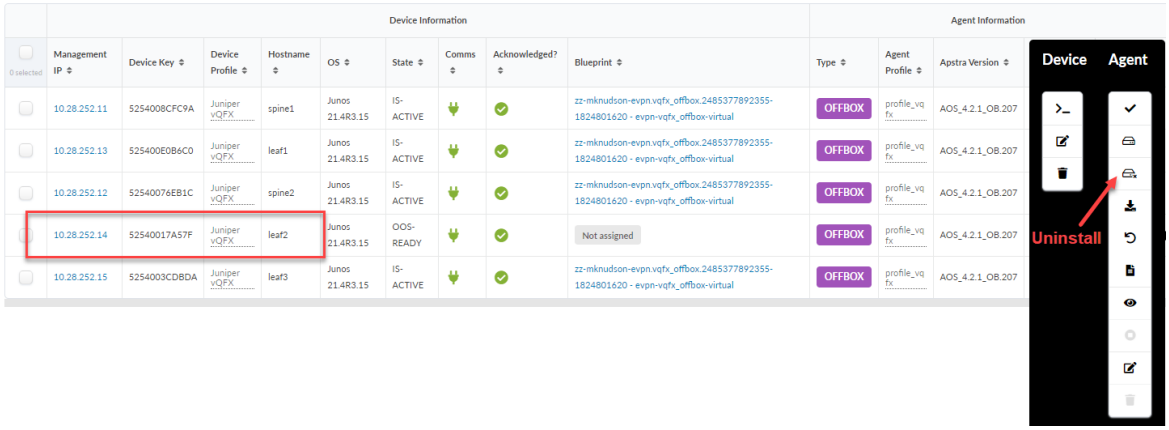


Leaf2 is now unassigned from the blueprint.



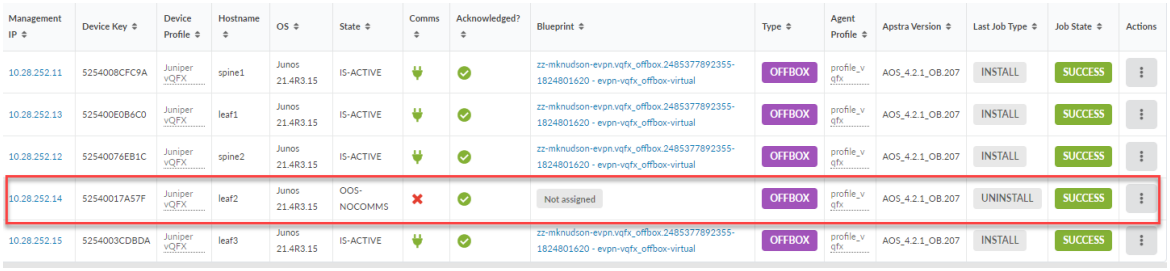
0 selected	Management IP	Device Key	Device Profile	Hostname	OS	State	Comms	Acknowledged?	Blueprint
	192.168.1.9						—		Not assigned
	10.28.252.11	5254008CFC9A	Juniper vQFX	spine1	Junos 21.4R3.15	IS-ACTIVE			zz-mknudson-evpn.vqfx_offbox.2485377892355-1824801620 - evpn-vqfx_offbox-virtual
	10.28.252.13	525400E0B6C0	Juniper vQFX	leaf1	Junos 21.4R3.15	IS-ACTIVE			zz-mknudson-evpn.vqfx_offbox.2485377892355-1824801620 - evpn-vqfx_offbox-virtual
	10.28.252.12	52540076EB1C	Juniper vQFX	spine2	Junos 21.4R3.15	IS-ACTIVE			zz-mknudson-evpn.vqfx_offbox.2485377892355-1824801620 - evpn-vqfx_offbox-virtual
	10.28.252.14	52540017A57F	Juniper vQFX	leaf2	Junos 21.4R3.15	OOS-READY			Not assigned

8. Uninstall the agent from leaf2 by clicking the **Uninstall** icon.



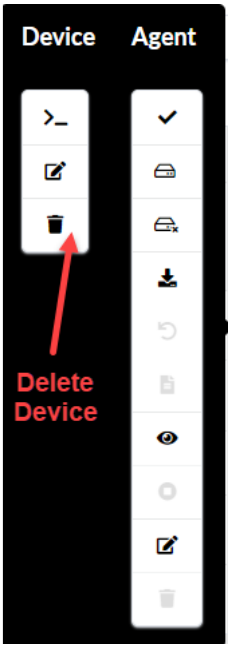
9. From the pop-up dialog box, click **Confirm** to uninstall the system agent.

10. After the uninstall completes, the leaf2 **State** shows as oos-nocomms with a red x in the **Comms** column.

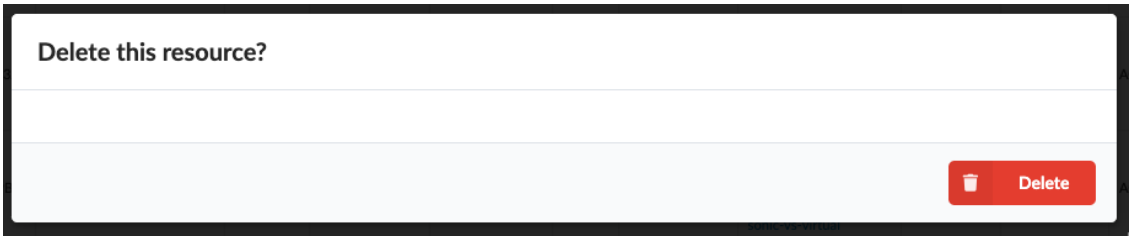


11. The final step is to delete the leaf2 device completely from Apstra. There are two steps:

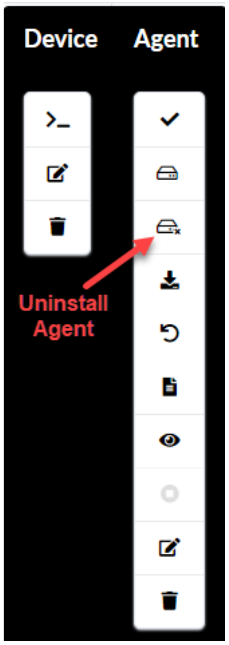
    a. First, click the trash icon in the **Device** panel.



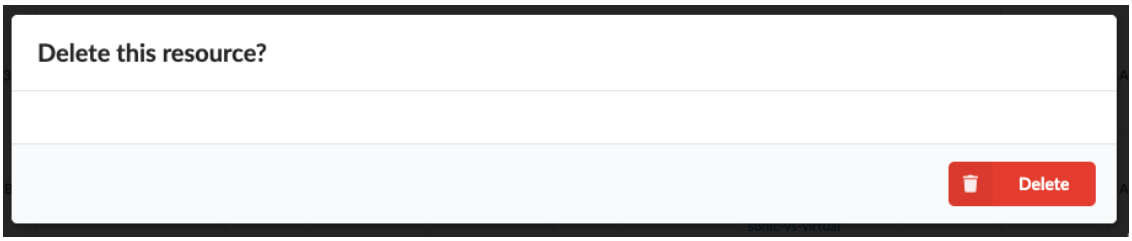
Then click **Delete** from the following dialog to remove the device.



b. The click the Agent **Uninstall** icon in the **Agent** panel.



Then click **Confirm** from the dialog box to uninstall the agent.



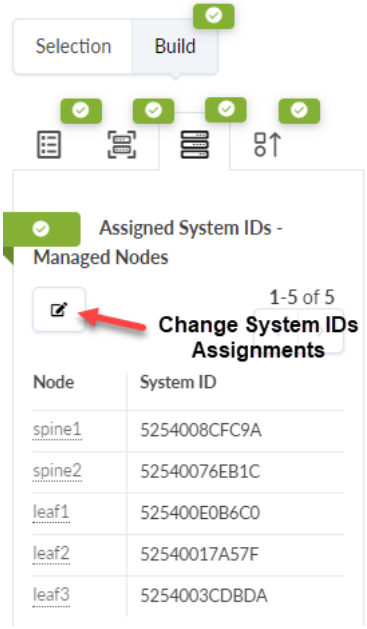
When complete the leaf2 device is removed from the blueprint.

Delete a Failed Leaf Device

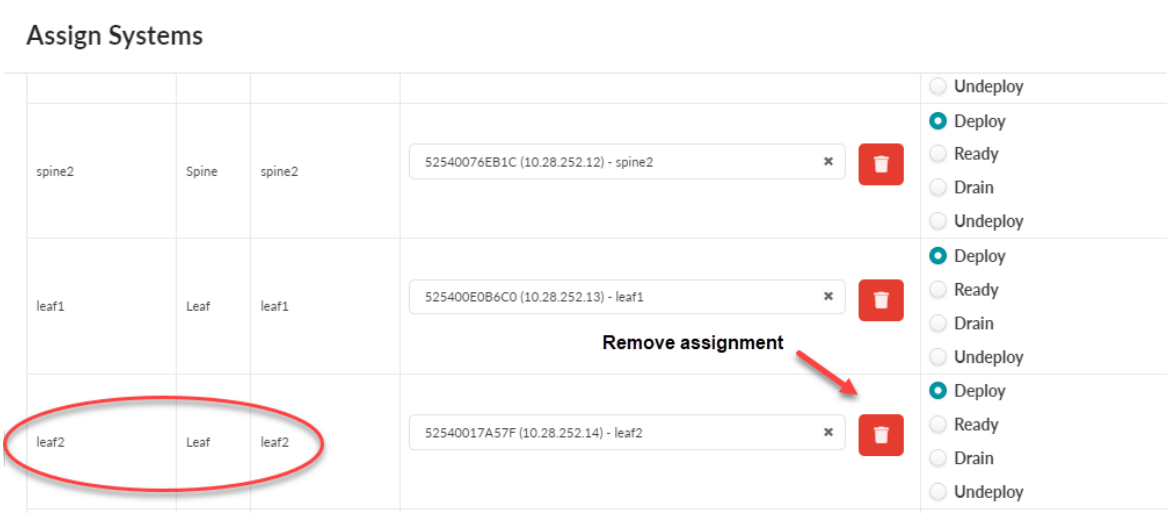
For this procedure, we'll delete a that has failed and is no longer communicating with Apstra.

1. From the left navigation menu in the Apstra GUI, navigate to **Blueprints> Staged > Physical > Build** and click the **Devices** icon. If the device already exists in the table, you can skip this step and proceed to Step 2.

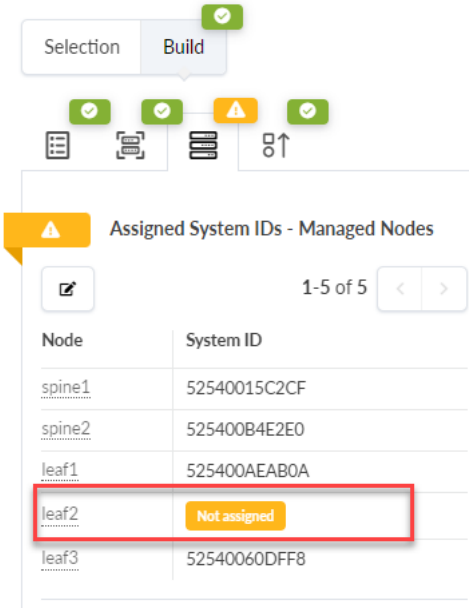
2. Click **Assigned System IDs - Managed Nodes >** and click the **Change System IDs Assignments** icon.



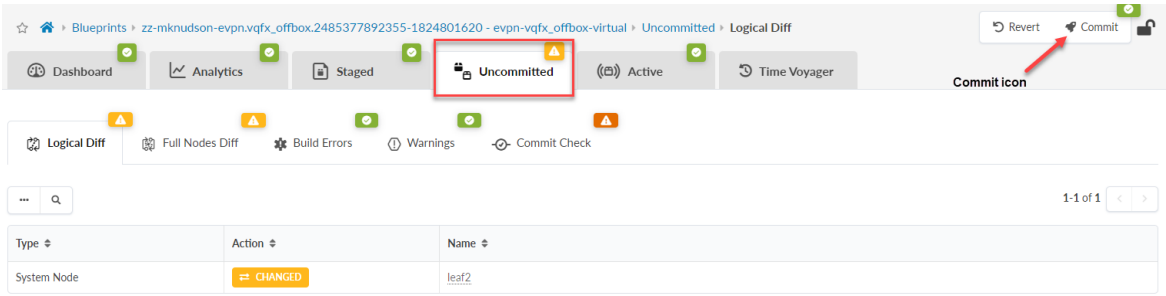
3. Click the trash icon to remove the leaf2 assignment, then click **Update Assignments**.



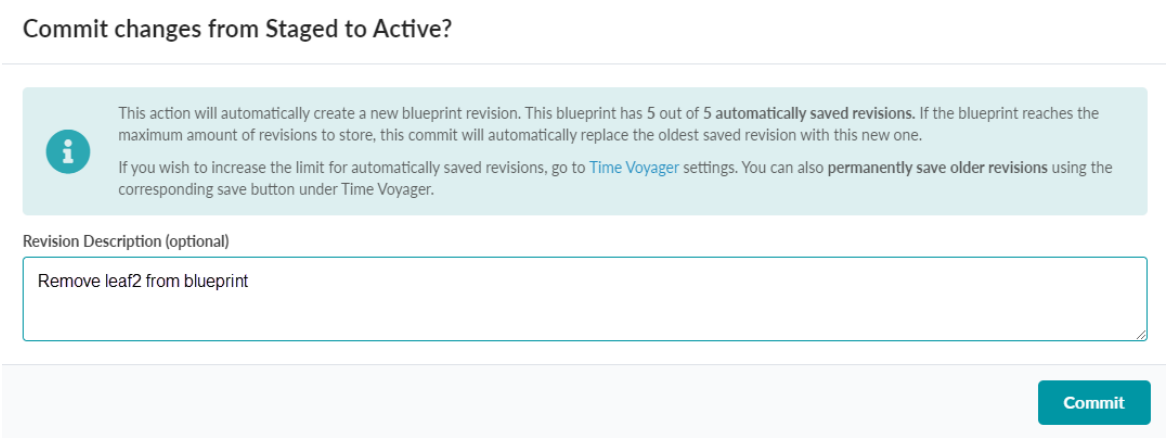
4. Verify that leaf2 is no longer assigned.



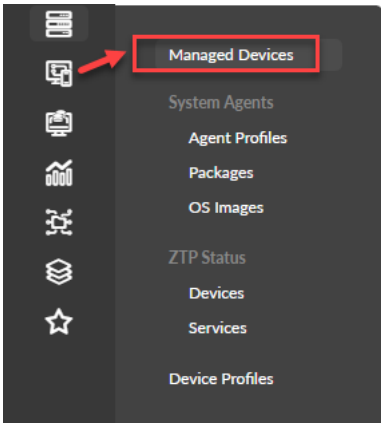
5. Navigate to the **Uncommitted** tab, then click the **Commit** rocket icon.



6. Add a comment if desired, and click **Commit**.



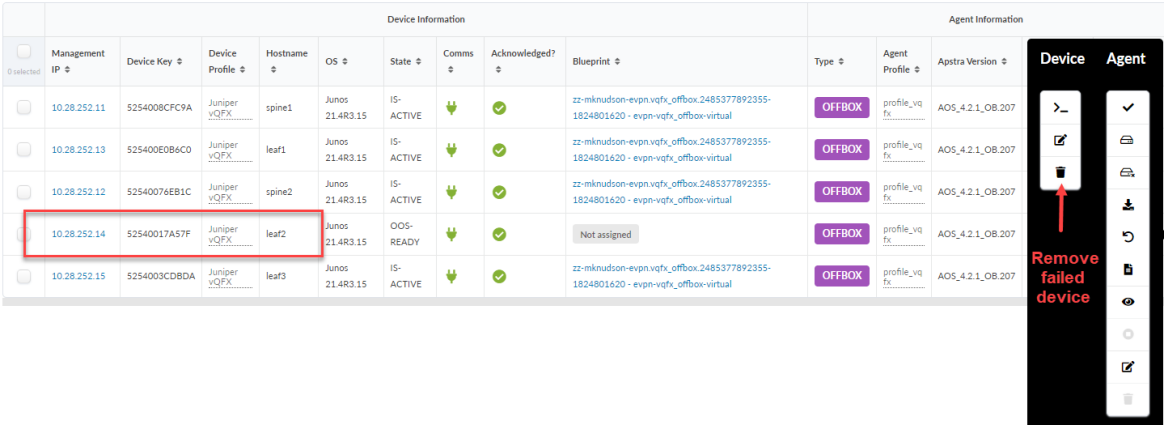
7. When the commit is complete, navigate to the **Devices > Managed Devices** panel to return to the managed devices table view.



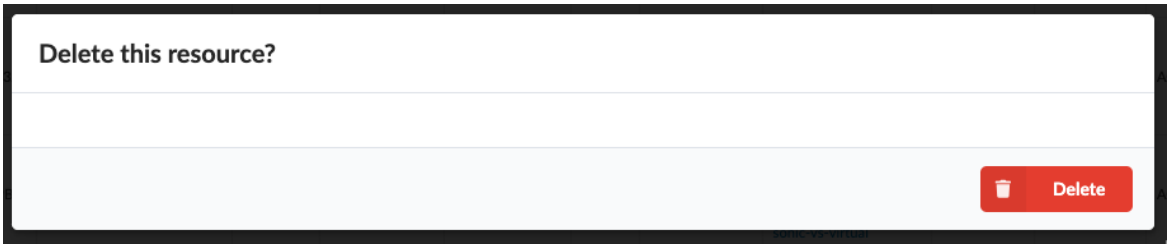
0 selected	Management IP	Device Key	Device Profile	Hostname	OS	State	Comms	Acknowledged?	Blueprint
	192.168.1.9								Not assigned
	10.28.252.11	5254008CFC9A	Juniper vQFX	spine1	Junos 21.4R3.15	IS-ACTIVE			zz-mknudson-evpn.vqfx_offbox.2485377892355-1824801620 - evpn-vqfx_offbox-virtual
	10.28.252.13	525400E0B6C0	Juniper vQFX	leaf1	Junos 21.4R3.15	IS-ACTIVE			zz-mknudson-evpn.vqfx_offbox.2485377892355-1824801620 - evpn-vqfx_offbox-virtual
	10.28.252.12	52540076EB1C	Juniper vQFX	spine2	Junos 21.4R3.15	IS-ACTIVE			zz-mknudson-evpn.vqfx_offbox.2485377892355-1824801620 - evpn-vqfx_offbox-virtual
	10.28.252.14	52540017A57F	Juniper vQFX	leaf2	Junos 21.4R3.15	OOS-READY			Not assigned

The Leaf2 device is now unassigned from the blueprint.

8. Click the trash icon in the **Device** panel.

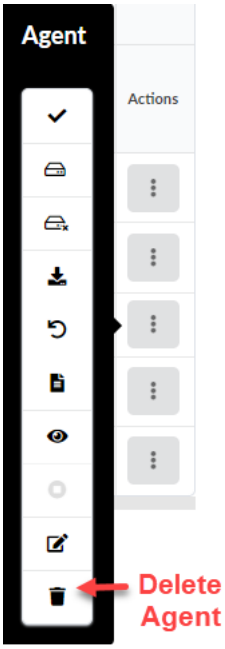


9. Click **Delete** to remove the failed device (leaf2) from the **Managed Devices** table.

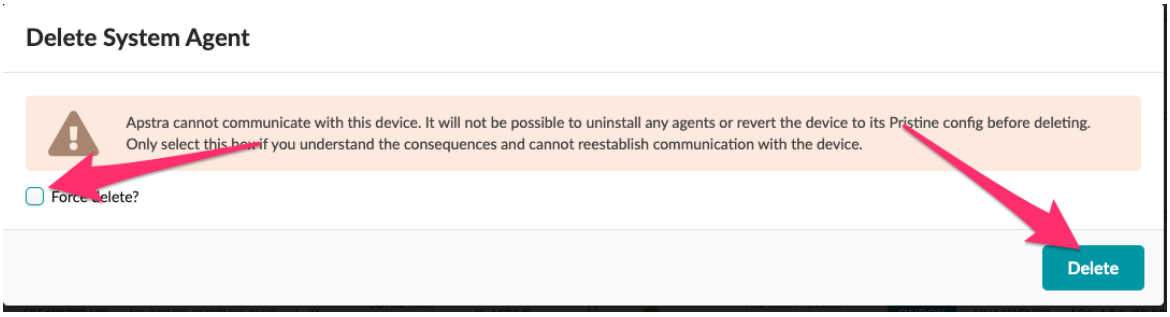


10. Delete the agent from the device.

    Select the device from the Managed Devices table (10.28.252.14 in our example) and click the trash icon in the Agent panel to delete the agent.



11. Click the Force delete check box, then click **Delete** to force the deletion.



You successfully removed the failed device from the blueprint.

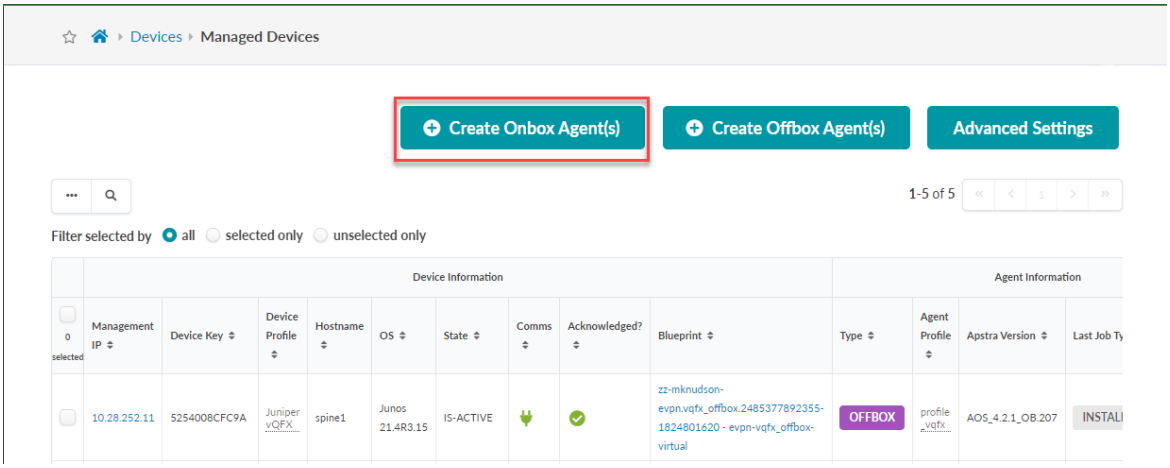
Add the Replacement Device to the Blueprint

Let's now take a look at what is required to replace the device back into the blueprint.

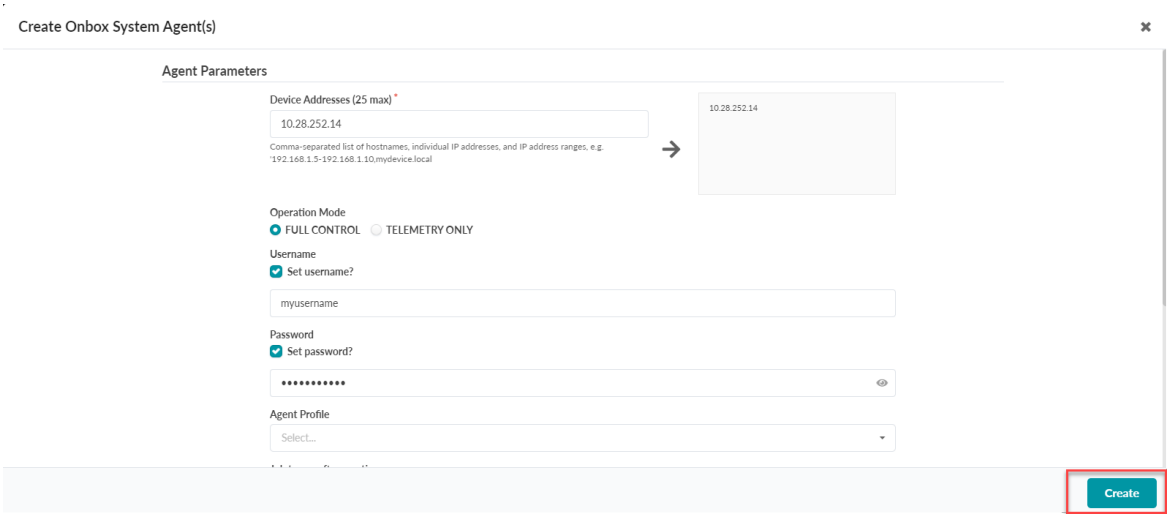
1. From the left navigation menu in the Apstra GUI, navigate to **Devices > Managed Devices**.

   If the device already exists in the table, you can skip this step and proceed to Step 2.

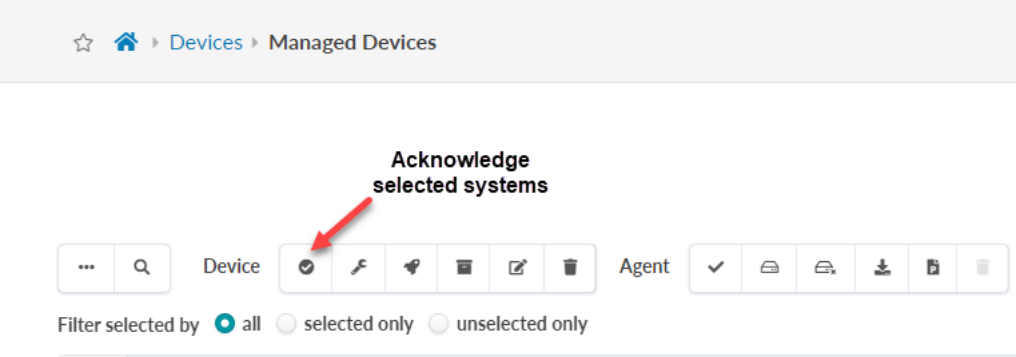
2. From the **Managed Devices** page, click **Create Onbox Agents(s)**. Note that SONIC devices use Onbox Agents.



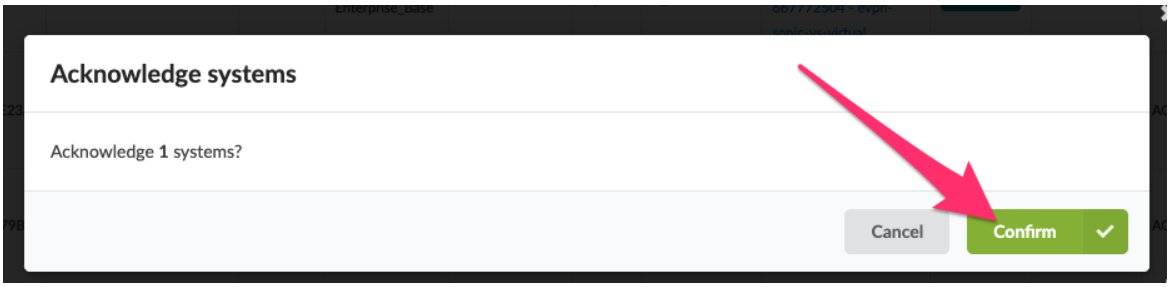
3. Enter the device address, username and password, then click **Create**.



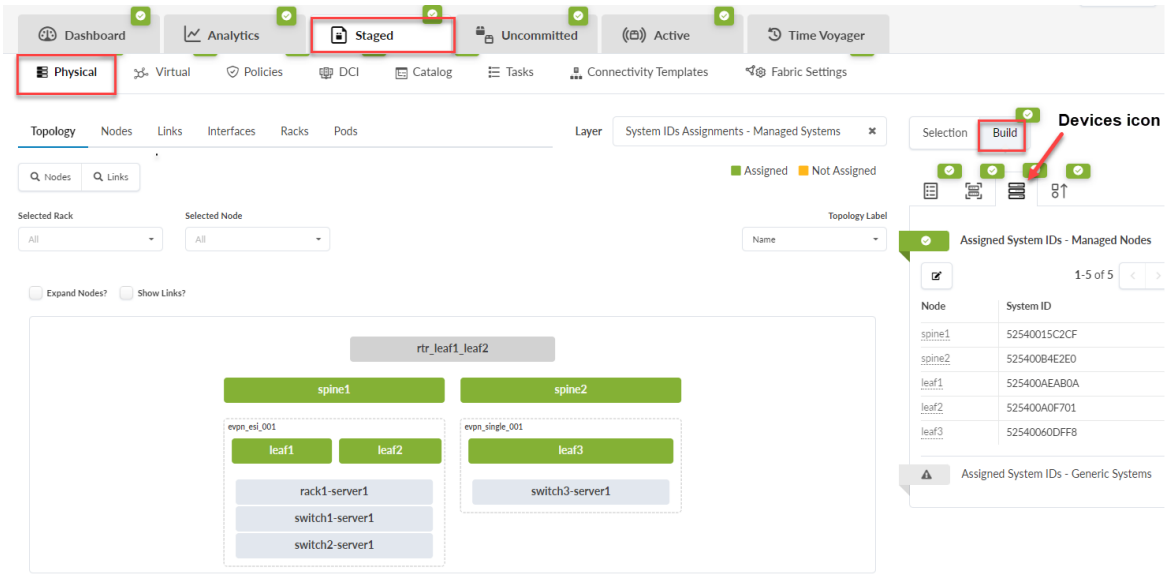
4. When the agent completes the installation, click **Acknowledge selected systems** from the Device panel located above the **Managed Devices** table.



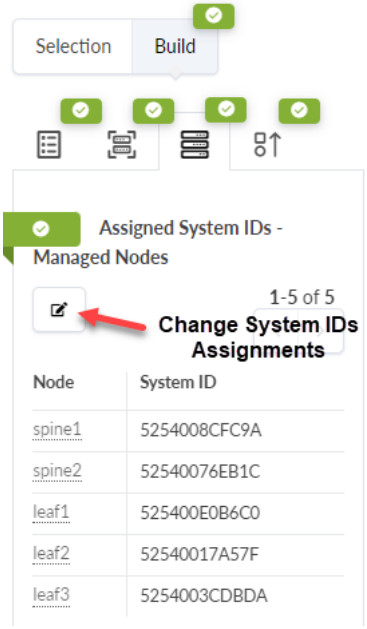
5. Click **Confirm** from the dialog box to acknowledge the device.



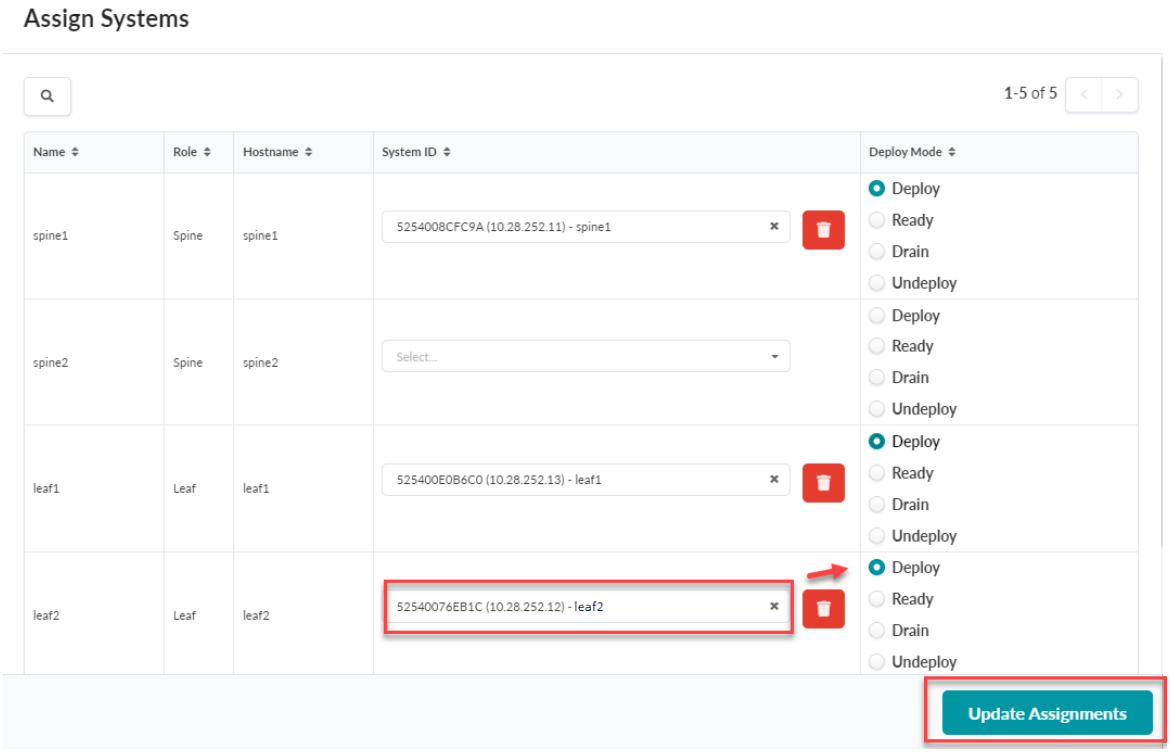
6. Navigate to **Blueprints > Blueprint ID > Staged > Physical > Build**, then click the **Devices** icon.



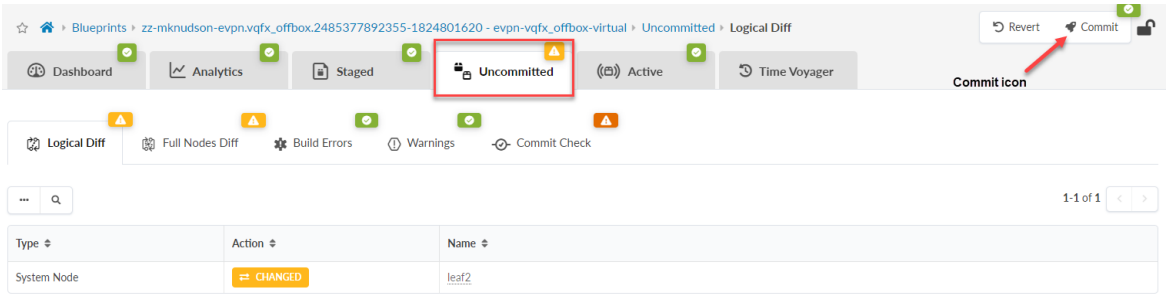
7. Click **Assigned System IDs - Managed Nodes** , then click the **Change System IDs Assignments** icon.



8. Select the leaf2 replacement device, click the **Deploy** radio button, then click **Update Assignments**.

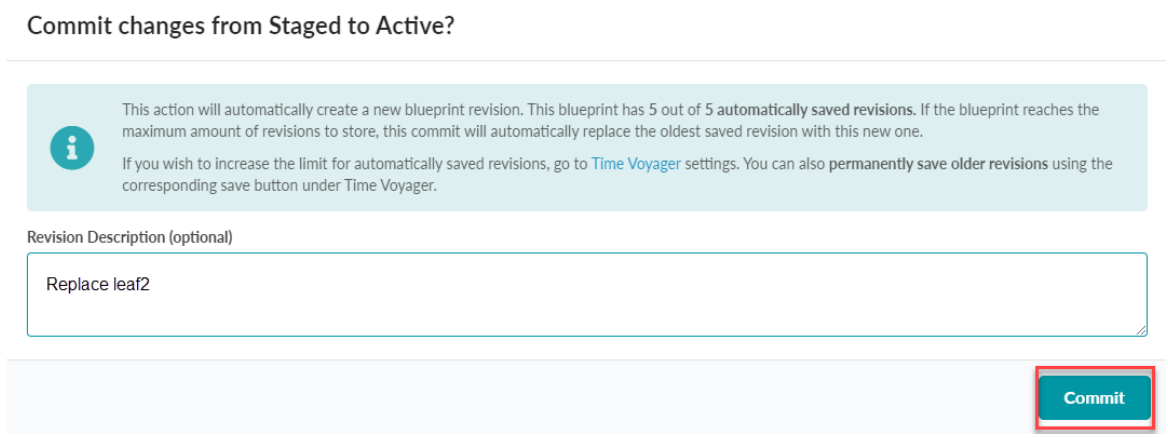


9. Commit the change by navigating to the **Uncommitted** tab, then click the **Commit** rocket icon.



10. Comment and click **Commit** to add the replacement device to your blueprint.

> **NOTE**: The deployment will take some time to stabilize and any anomalies should eventually converge and disappear.



Replace a Device in Apstra Using Apstra Terraform Provider

You can easily replace a device in a blueprint using Apstra Terraform Provider. The below diagram shows the topology we are using in this procedure.

In the following example Terraform configuration, we are replacing the leaf device (`rack_a_001_leaf1`).

NOTE: We assume that the Terraform configuration is applied to the Apstra server and all of the resources for the blueprint are allocated and committed.

Example Terraform Configuration

```
terraform {
  required_providers {
    apstra = {
      source = "Juniper/apstra"
    }
  }
}

provider "apstra" {
  url                     = "https://admin:admin@10.28.24.3:443"
  tls_validation_disabled = true
  blueprint_mutex_enabled = false
  experimental = true #New Apstra 4.2 API
}

resource "apstra_rack_type" "a_rack" {
  name  = "rack_a"
  leaf_switches = {
    leaf1 = {
      logical_device_id = "slicer-7x10-1"
      spine_link_speed = "10G"
      spine_link_count = 1
    }
  }
  fabric_connectivity_design = "l3clos"
}
resource "apstra_template_rack_based" "a_template" {
  name = "template_a"
  asn_allocation_scheme = "unique"
  overlay_control_protocol = "evpn"
  spine = {
    logical_device_id = "slicer-7x10-1"
    count = 2
  }
```

```
  rack_infos = {
    (apstra_rack_type.a_rack.id) = { count = 1}
  }
}
resource "apstra_datacenter_blueprint" "a_blueprint" {
  name = "blueprint_a"
  template_id = apstra_template_rack_based.a_template.id
}



resource "apstra_managed_device_ack" "spine1" {
  agent_id = "e44df49e-556b-446e-a02f-7fd23804901e"
  device_key = "525400BBC20C"
}
resource "apstra_managed_device_ack" "spine2" {
  agent_id = "b5726b1d-29ba-4ab0-ac76-5def33291cc4"
  device_key = "525400AF79BA"
}
resource "apstra_managed_device_ack" "leaf1" {
  agent_id = "8f448798-c260-46b7-b7fa-46609f0598d4"
  device_key = "525400E9E2FF"
}
resource "apstra_datacenter_device_allocation" "spine1" {
  blueprint_id = apstra_datacenter_blueprint.a_blueprint.id
  node_name = "spine1"
  device_key = apstra_managed_device_ack.spine1.device_key
  deploy_mode              = "deploy"
}
resource "apstra_datacenter_device_allocation" "spine2" {
  blueprint_id = apstra_datacenter_blueprint.a_blueprint.id
  node_name = "spine2"
  device_key = apstra_managed_device_ack.spine2.device_key
  deploy_mode              = "deploy"
}
resource "apstra_datacenter_device_allocation" "leaf1" {
  blueprint_id = apstra_datacenter_blueprint.a_blueprint.id
  node_name = "rack_a_001_leaf1"
  device_key = apstra_managed_device_ack.leaf1.device_key
  deploy_mode              = "deploy"
}
```

To replace a device using Terraform:

1. Remove the following resource block for the device from the Terraform configuration. Note that you cannot remove a device from Apstra without first removing it from your blueprint.

   In this example, we are removing the `rack_a_001_leaf1` device.

```
    resource "apstra_datacenter_device_allocation" "leaf1" {
    blueprint_id = apstra_datacenter_blueprint.a_blueprint.id
    node_name = "rack_a_001_leaf1"
    device_key = apstra_managed_device_ack.leaf1.device_key
    deploy_mode              = "deploy"
    }
```

   Run the `terraform apply` command to apply the changes.

```
user@user-mbp terraform % terraform apply
    apstra_datacenter_device_allocation.leaf1: Refreshing state...
    apstra_managed_device_ack.spine2: Refreshing state...
    apstra_managed_device_ack.leaf1: Refreshing state...
    apstra_managed_device_ack.spine1: Refreshing state...
    apstra_rack_type.a_rack: Refreshing state... [id=wjdosvpvs8qdkyecs2slrw]
    apstra_template_rack_based.a_template: Refreshing state... [id=3fcd246d-f7eb-4850-ba18-
bd8429aa3c8d]
    apstra_datacenter_blueprint.a_blueprint: Refreshing state... [id=9c2404f6-
a875-4d29-9add-919601601b04]
    apstra_datacenter_device_allocation.spine2: Refreshing state...
    apstra_datacenter_device_allocation.spine1: Refreshing state...

    Terraform used the selected providers to generate the following execution plan. Resource
actions are indicated with the following symbols:
      - destroy

    Terraform will perform the following actions:

      # apstra_datacenter_device_allocation.leaf1 will be destroyed
      # (because apstra_datacenter_device_allocation.leaf1 is not in configuration)
      - resource "apstra_datacenter_device_allocation" "leaf1" {
          - blueprint_id             = "9c2404f6-a875-4d29-9add-919601601b04" -> null
          - deploy_mode              = "deploy" -> null
          - device_key               = "525400E9E23A" -> null
          - device_profile_node_id   = "sfBQVnyC5MBMtWoYLpU" -> null
          - initial_interface_map_id = "VS_SONiC_BUZZNIK_PLUS__slicer-7x10-1" -> null
          - interface_map_name       = "VS_SONiC_BUZZNIK_PLUS__slicer-7x10-1" -> null
```

```
      - node_id                  = "v4U3_ek6bH3t_ppYEXA" -> null
      - node_name                = "rack_a_001_leaf1" -> null
    }

Plan: 0 to add, 0 to change, 1 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

apstra_datacenter_device_allocation.leaf1: Destroying...
apstra_datacenter_device_allocation.leaf1: Destruction complete after 4s

Apply complete! Resources: 0 added, 0 changed, 1 destroyed.
```

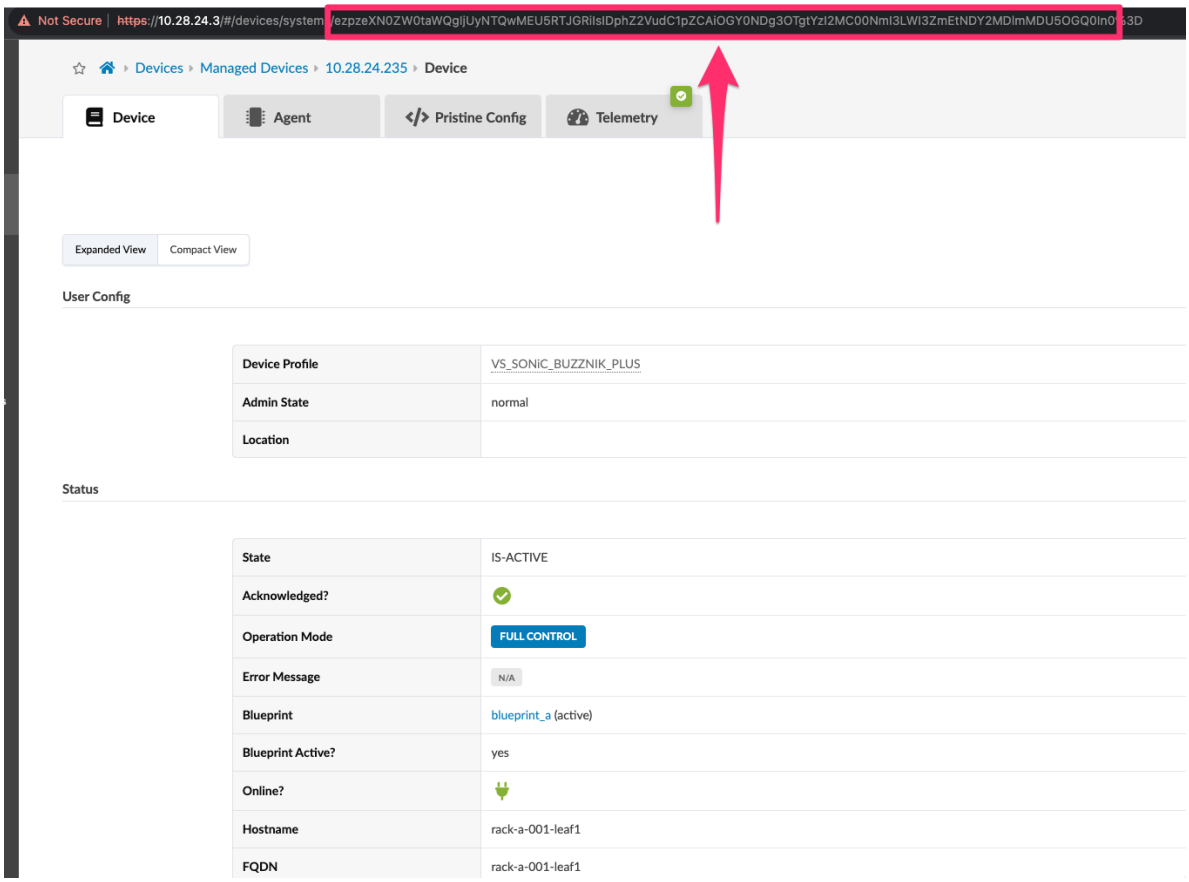
2. After the apply is complete, commit the changes.
3. Update the Terraform configuration with the new information for the replacement device.

   Replace the `device_key` and the `agent_id` with the new values.

```
resource "apstra_managed_device_ack" "leaf1" { agent_id = "8f448798-c260-46
b7-b7fa-46609f0598d4" device_key = "525400E9E2FF" }
```

   You can derive these values in the Apstra GUI by navigating to the **Devices > Managed Devices** page and selecting the device IP address as follows:

4. From the **Device** page, copy the encoded device key and agent ID from the URL string.



To decode the URL string and obtain the new `device_key` (`system-id`) and agent ID values, run the `pbpaste | base64 --decode` command. For example:

```
pbpaste | base64 --decode
    {:system-id "525400E9E2FF", :agent-id "8f448798-c260-46b7-b7fa-46609f0598d4%
```

> **NOTE**: You can also derive the agent-id from a data source, from Swagger, or from the Terraform resource that created it.

5. Once you have replaced the information into Terraform, apply the new configuration:

```
user@user-mbp terraform % terraform apply
    apstra_managed_device_ack.spine1: Refreshing state...
    apstra_managed_device_ack.spine2: Refreshing state...
    apstra_rack_type.a_rack: Refreshing state... [id=wjdosvpvs8qdkyecs2slrw]
    apstra_template_rack_based.a_template: Refreshing state... [id=3fcd246d-f7eb-4850-ba18-
```

```
bd8429aa3c8d]
    apstra_datacenter_blueprint.a_blueprint: Refreshing state... [id=9c2404f6-
a875-4d29-9add-919601601b04]
    apstra_datacenter_device_allocation.spine2: Refreshing state...
    apstra_datacenter_device_allocation.spine1: Refreshing state...

    Terraform used the selected providers to generate the following execution plan. Resource
actions are indicated with the following symbols:
      + create

    Terraform will perform the following actions:

      # apstra_datacenter_device_allocation.leaf1 will be created
      + resource "apstra_datacenter_device_allocation" "leaf1" {
          + blueprint_id             = "9c2404f6-a875-4d29-9add-919601601b04"
          + deploy_mode              = "deploy"
          + device_key               = "525400E9E2FF"
          + device_profile_node_id   = (known after apply)
          + initial_interface_map_id = (known after apply)
          + interface_map_name       = (known after apply)
          + node_id                  = (known after apply)
          + node_name                = "rack_a_001_leaf1"
        }

      # apstra_managed_device_ack.leaf1 will be created
      + resource "apstra_managed_device_ack" "leaf1" {
          + agent_id   = "8f448798-c260-46b7-b7fa-46609f0598d4"
          + device_key = "525400E9E2FF"
          + system_id  = (known after apply)
        }

    Plan: 2 to add, 0 to change, 0 to destroy.

    Do you want to perform these actions?
      Terraform will perform the actions described above.
      Only 'yes' will be accepted to approve.

      Enter a value: yes

    apstra_managed_device_ack.leaf1: Creating...
    apstra_managed_device_ack.leaf1: Creation complete after 1s
    apstra_datacenter_device_allocation.leaf1: Creating...
    apstra_datacenter_device_allocation.leaf1: Creation complete after 7s
```

```
Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
```

6. Commit your changes in the Apstra GUI or in Terraform to finish replacing the device.

Good job! You successfully replaced a device in Apstra.

NOTE: In our examples, we removed any existing configuration from the new device and performed a device reboot. However, in some use cases, such as SONiC devices, you might be required to remove any existing configuration from the device.

For example:

```
root@sonic:/home/admin# rm /etc/sonic/config_db.json
    root@sonic:/home/admin#  config-setup factory
    root@sonic:/home/admin# reboot
```

See the Juniper Apstra User Guide for information about configuring a SONiC device.