vLABの登録/利用ガイド

Juniper Networks

2025/4

JUNIPER NETWORKS | Engineering Simplicity

はじめに

本資料は、リモートラボ環境であるJuniper vLabsを利用*するための

登録ガイドとなります。

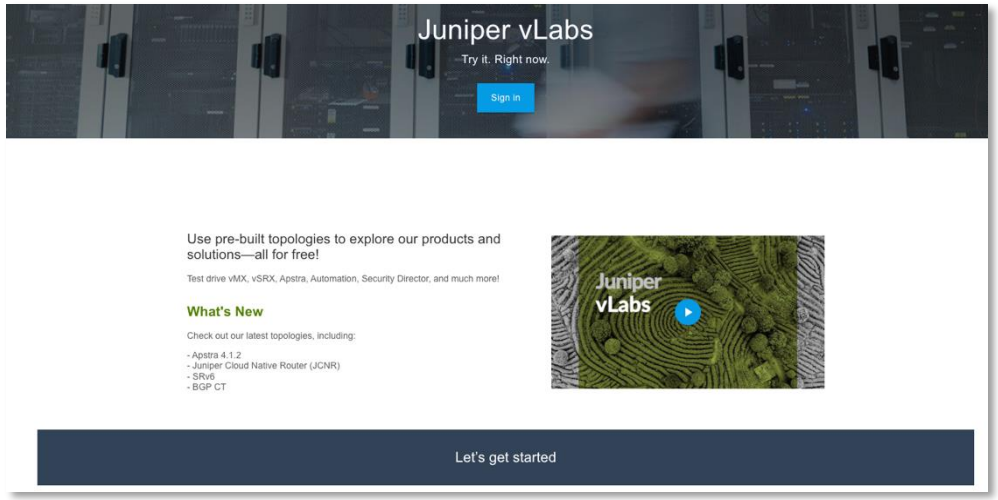
2025/4 ガイドとなるため、本資料とは動作が異なる場合もあるため、ご了承ください。

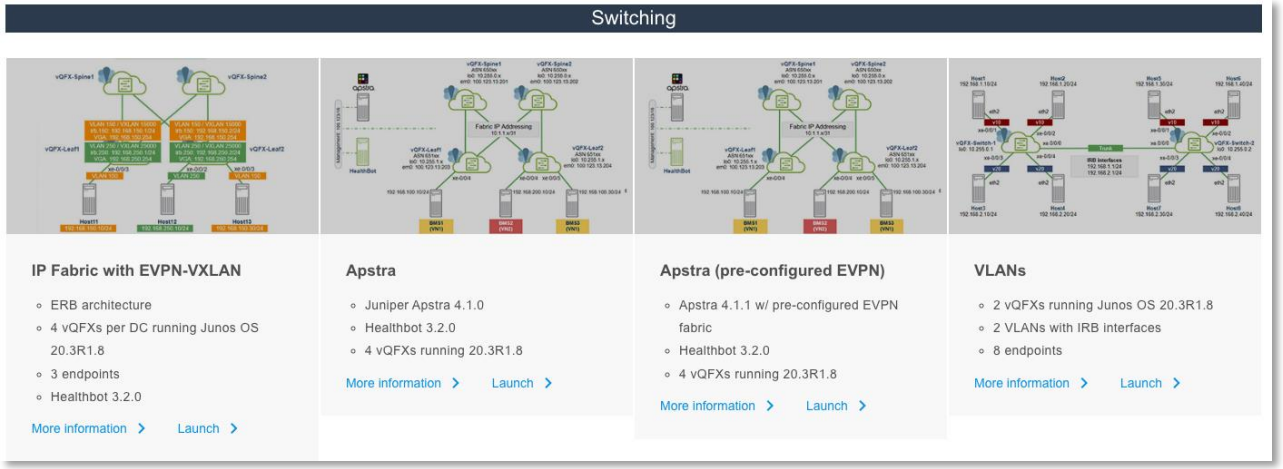
vLAB

vLabはジュニパーの提供するユースケースや機能をテストできる*Webベースのプラットフォームです。

ジュニパーの複数の製品ラインを利用できるvLabsは、ルーティング、スイッチング、セキュリティ、自動化、テレメトリ、など様々な環境を仮想環境としてご使用いただけます。

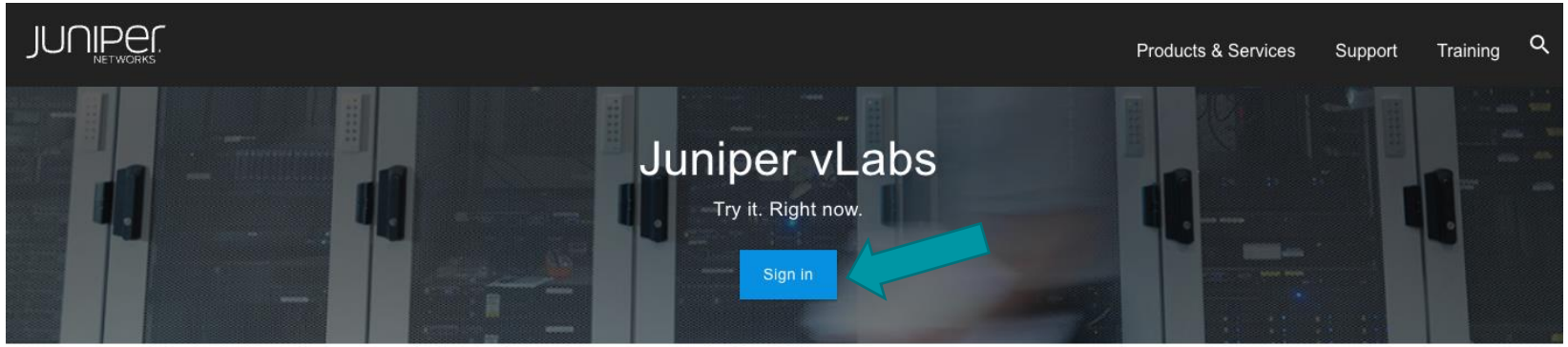
vLabsセッションはアカウントを登録いただいた方は無料でご利用いただけます。





*仮想環境・仮想アプライアンスの為環境による機能等の制限がございます

vLAB登録手順 (1)



Use pre-built topologies to explore our products and solutions—all for free!

Test drive vMX, vSRX, Apstra, Automation, Security Director, and much more!

What's New

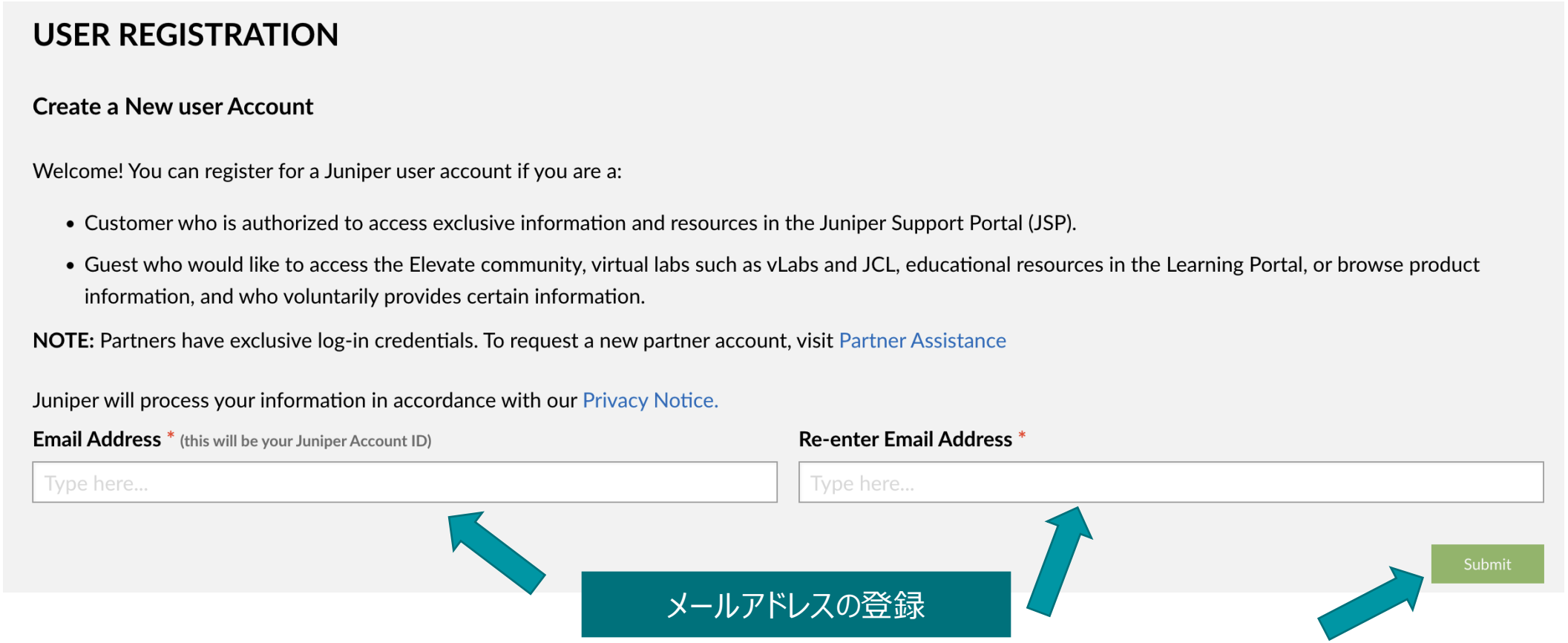
Check out our latest topologies, including:

- Apstra 4.1.2
- Juniper Cloud Native Router (JCNR)
- SRv6
- BGP CT

https://jlabs.juniper.net/vlabs/

vLAB登録手順 (2)

vLAB登録手順 (3)

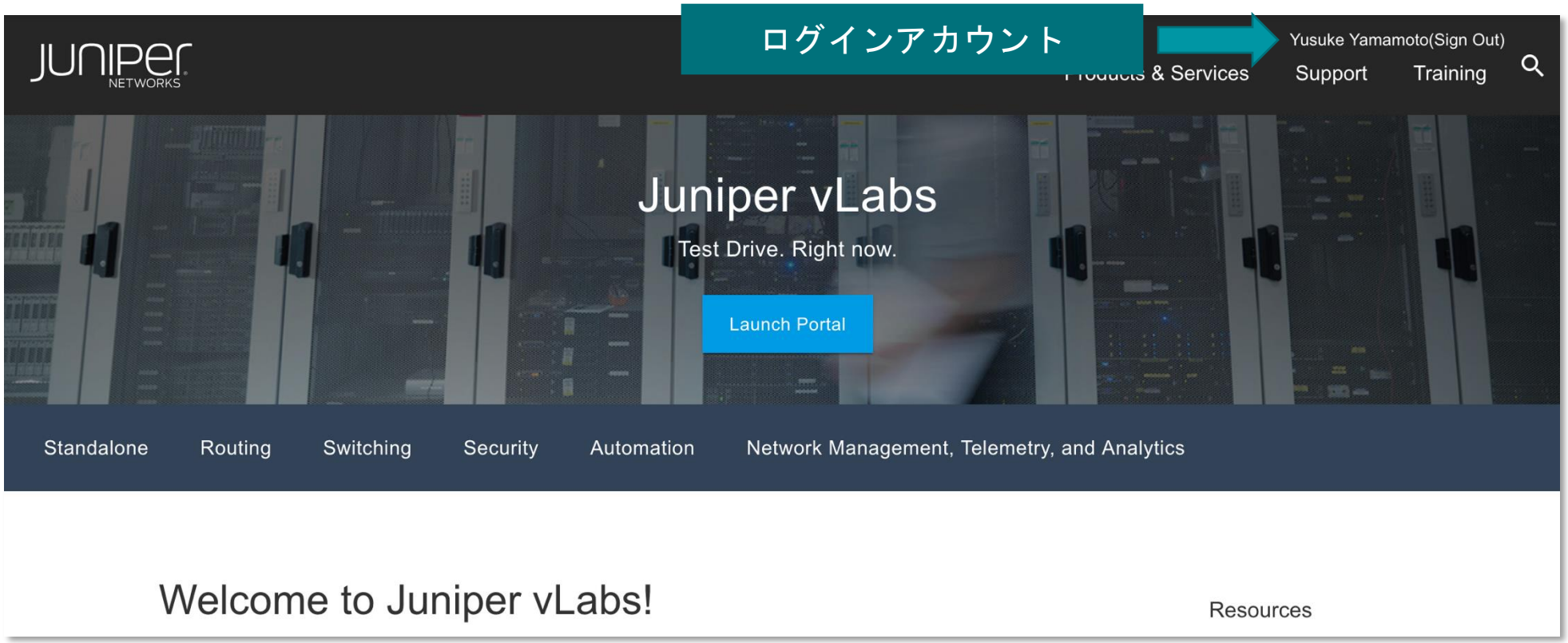


メールアドレスに登録のためのURLが届きますので指示に沿って登録を進めてください

手順内では「Access to EngNet, vLabs & other JCL Tools」を選択ください

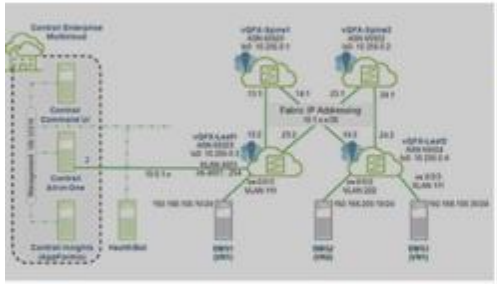
vLAB登録手順 (4)

vLAB利用手順 (1)



vLAB利用手順 (2)

Network Management, Telemetry, and Analytics

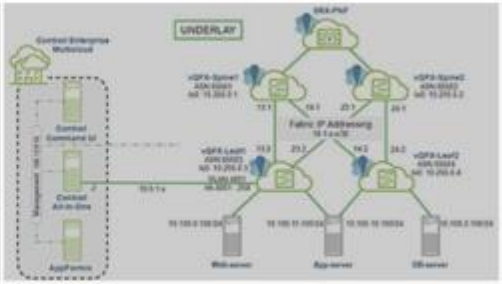


Contrail Enterprise Multicloud

- Contrail 2003 with Contrail Command UI
- Contrail Insights (AppFormix) 3.1.15
- 4 vQFXs running Junos OS 18.1R3-S5
- BMSs running CentOS 7.4

More information >

Launch >

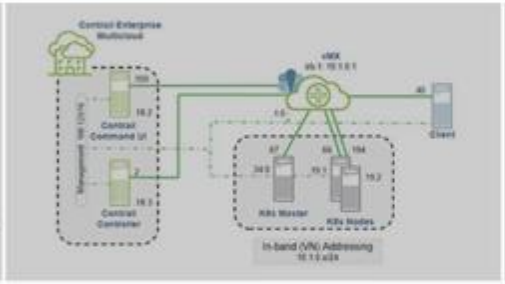


Contrail Enterprise Multicloud with PNF

- Contrail 1909 with Contrail Command UI; Contrail Insights (AppFormix) 3.1.2
- 4 vQFXs running Junos OS 18.1R3-S5
- vSRX providing PNF functionality
- BMSs running CentOS 7.4

More information >

Launch >

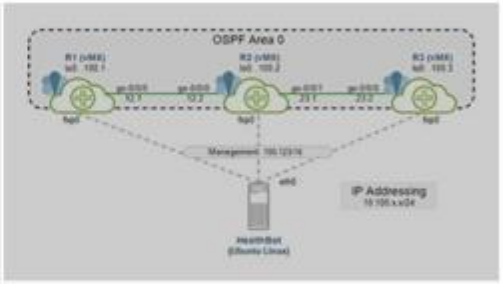


Contrail Enterprise Multicloud with Kubernetes

- Contrail 1910 with Contrail Command UI
- 1 vMX running Junos OS 18.1R3-S5
- Kubernetes 1.12

More information >

Launch >



HealthBot

- Healthbot 3.0.0 on Linux server
- 3 vMXs running Junos OS 18.3R1.9

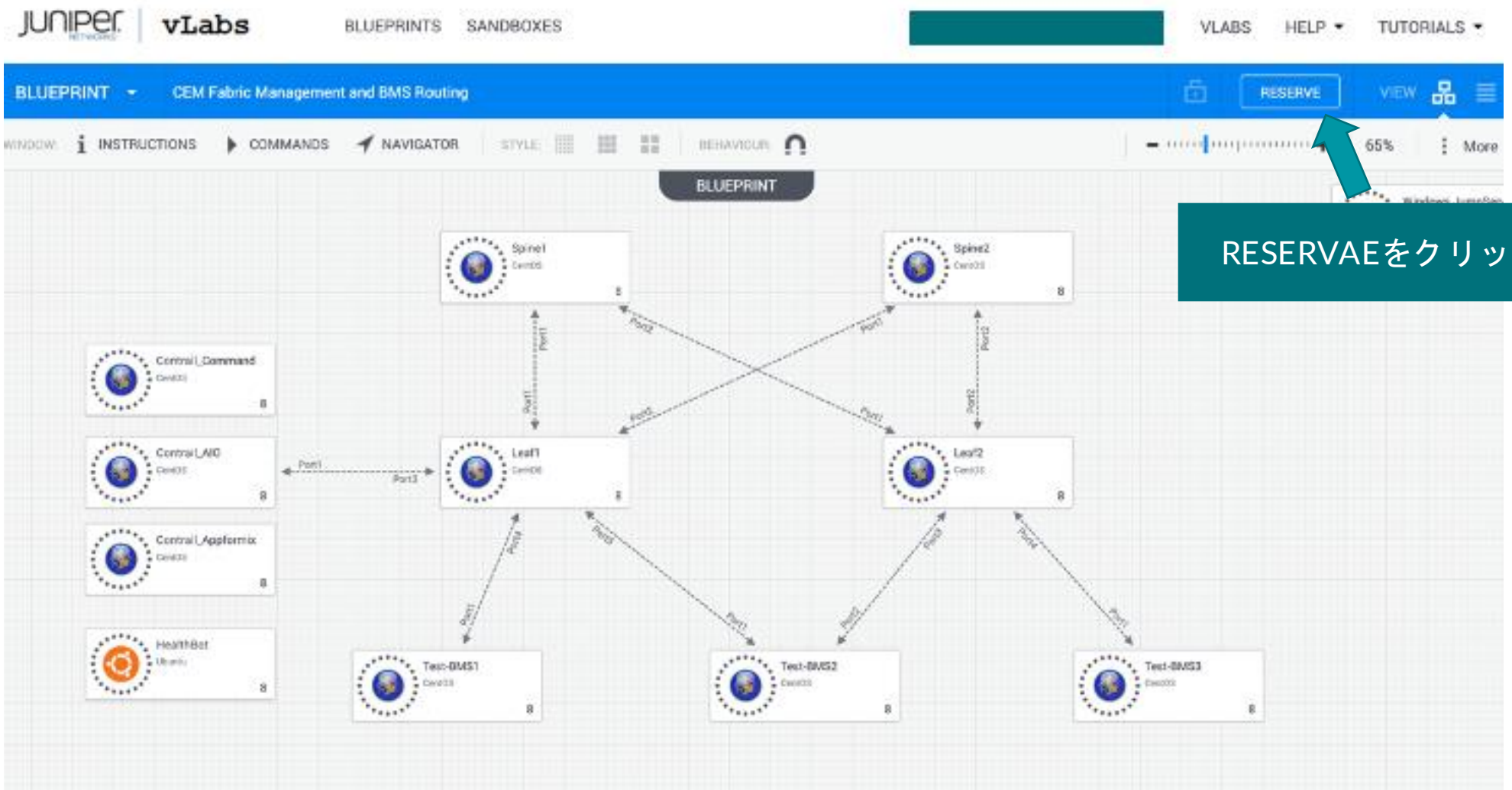
More information >

Launch >

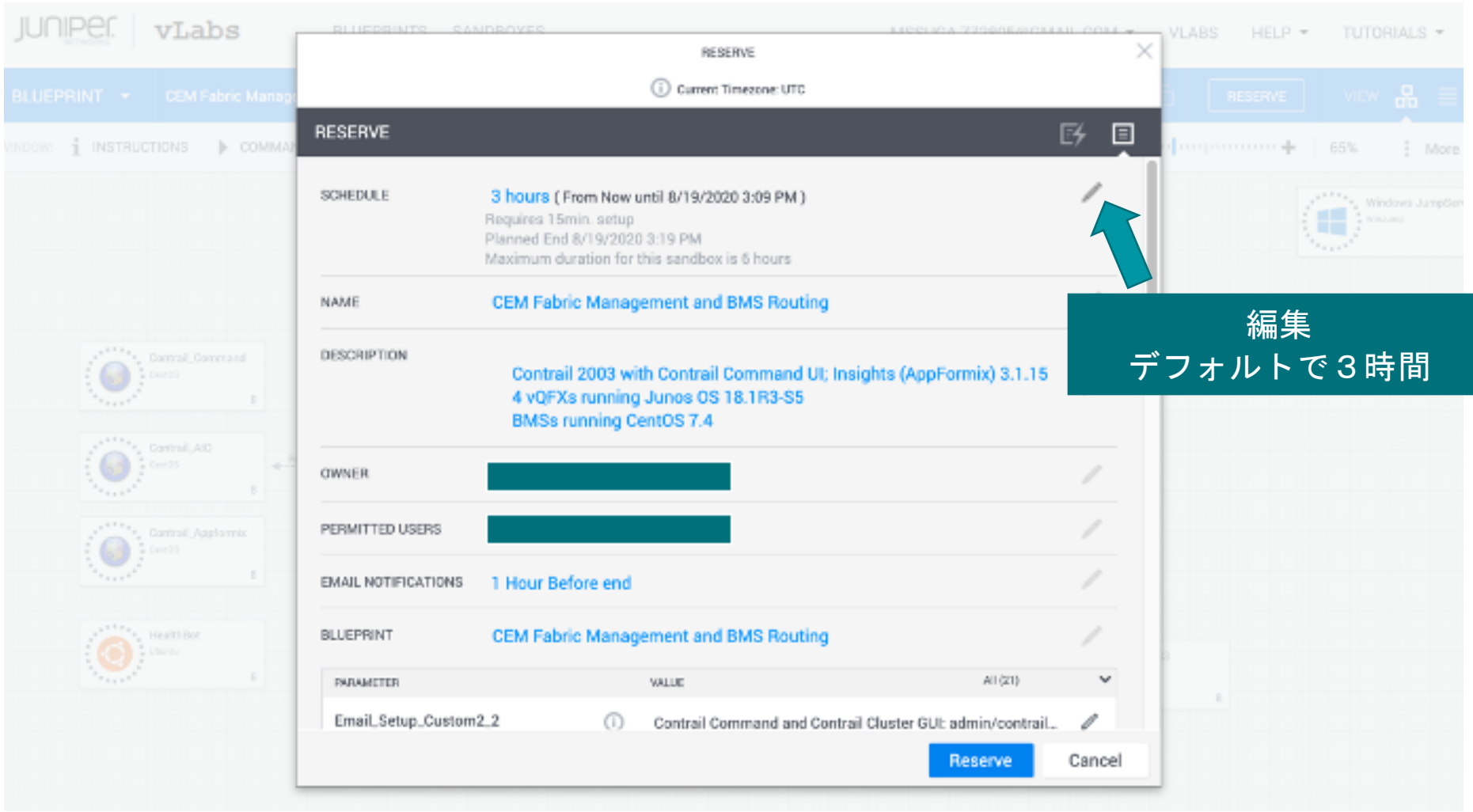
VLAB開始

VLABの構成等の情報確認

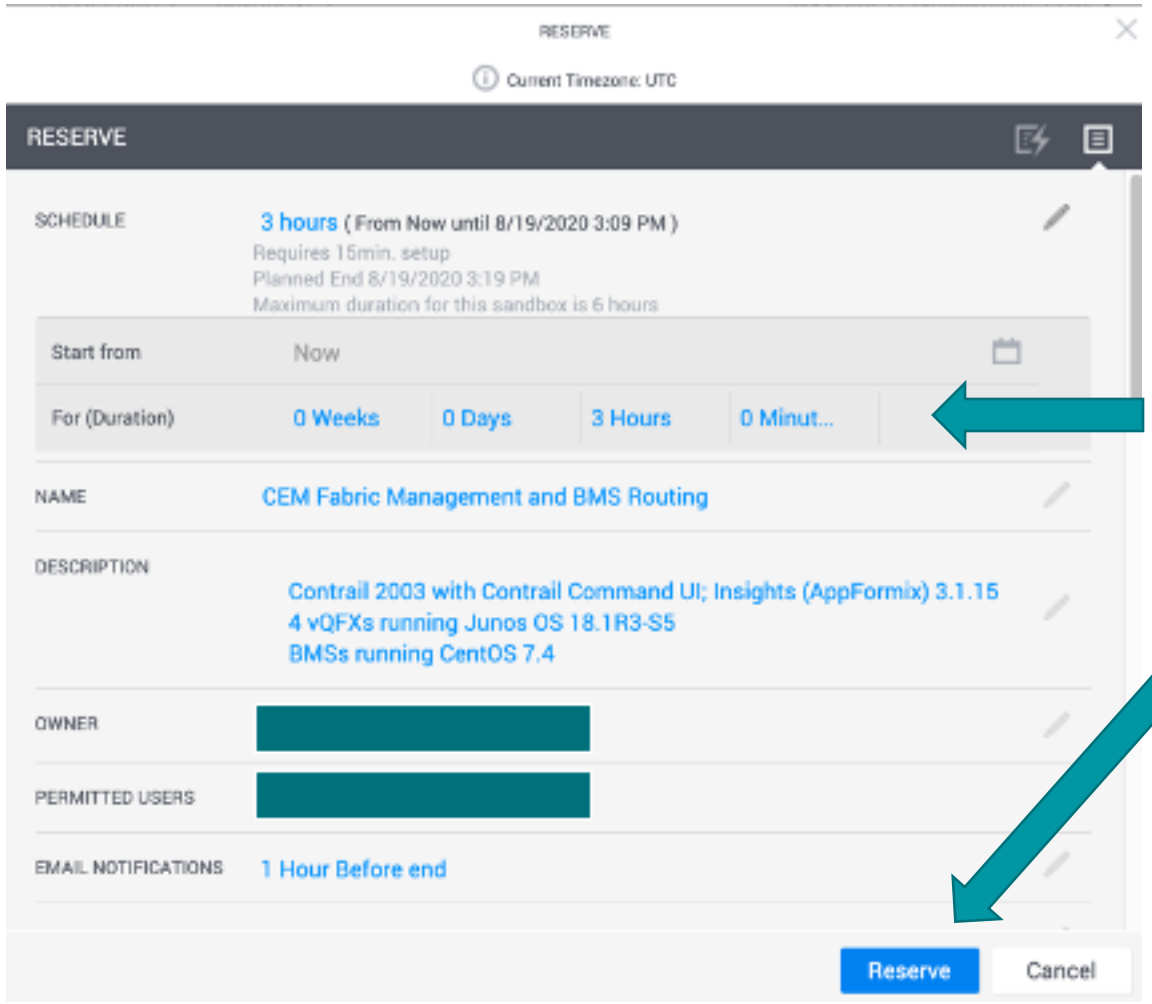
vLAB利用手順 (3)



vLAB利用手順 (4)

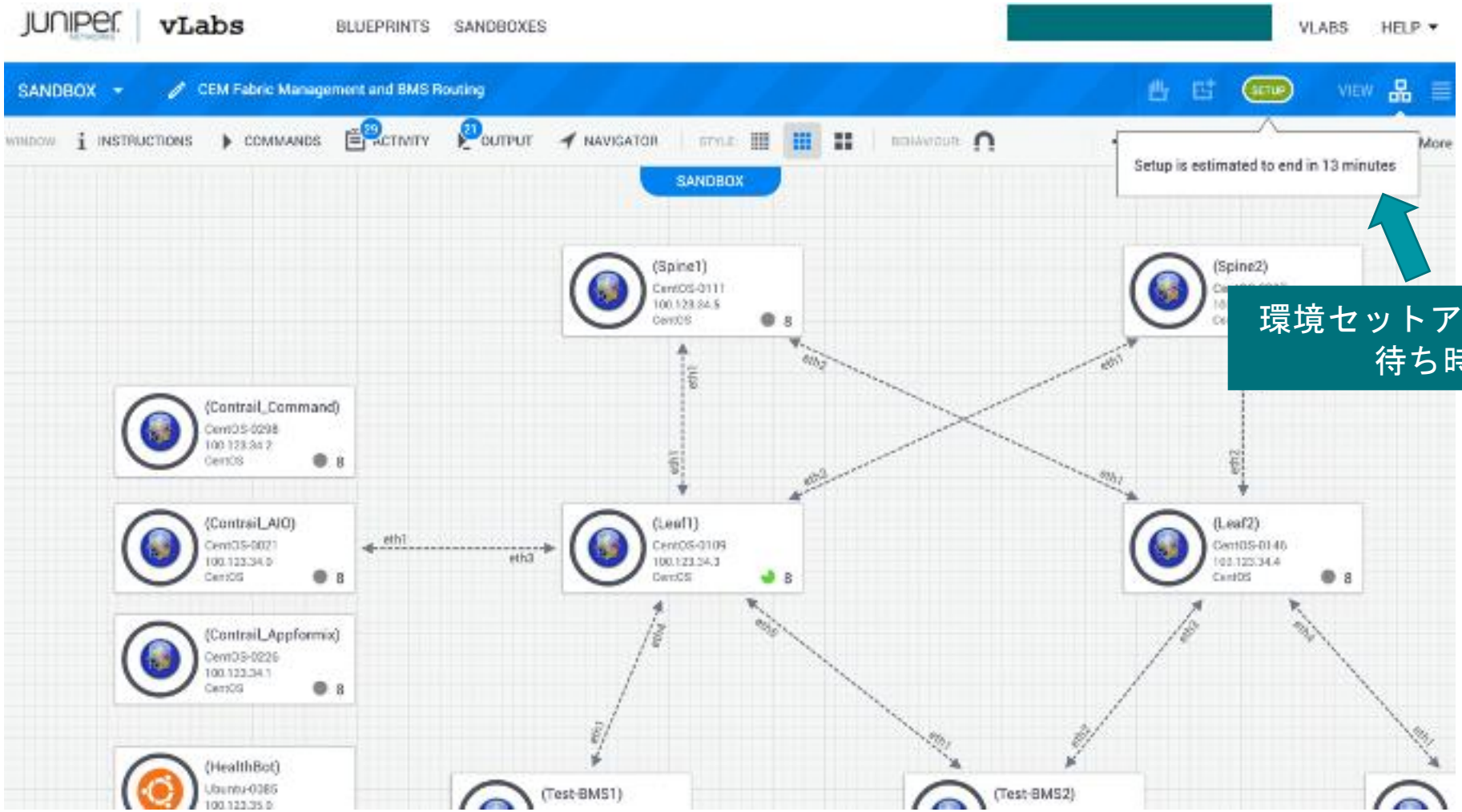


vLAB利用手順 (5)

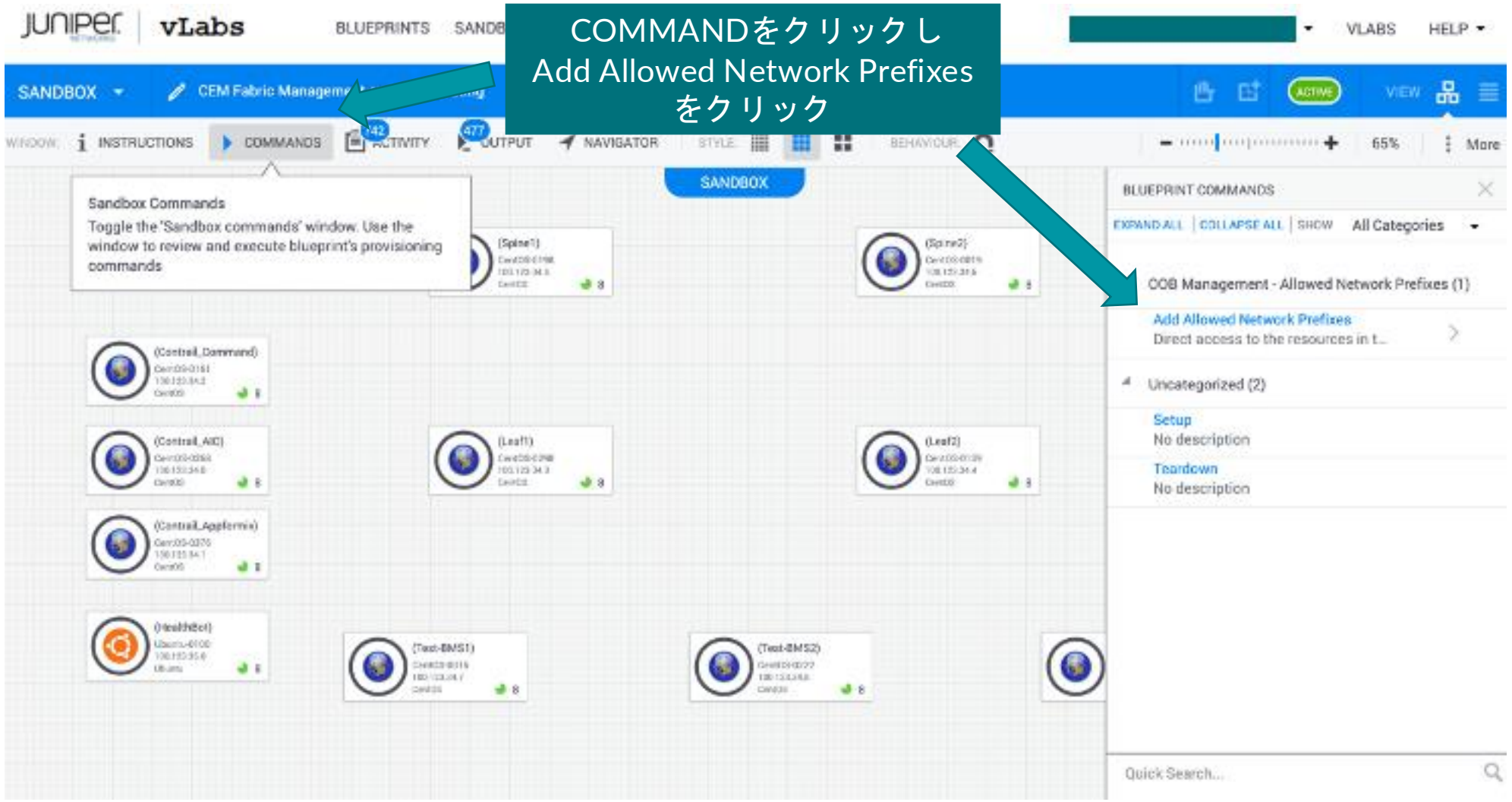


VLABの稼働時間を設定

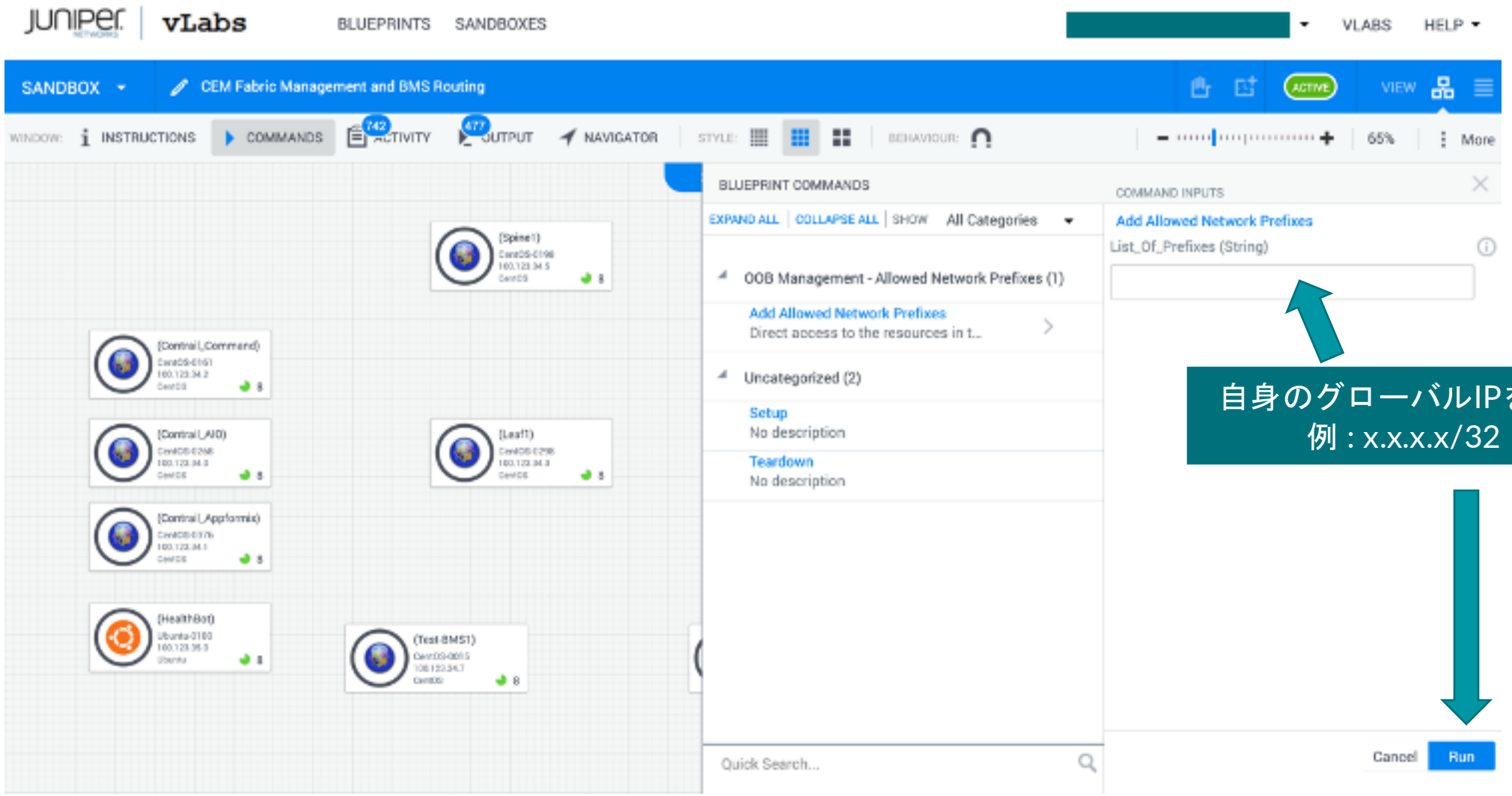
vLAB利用手順 (6)



vLAB利用手順 (7)



vLAB利用手順 (8)



vLAB利用手順 (9)

登録したメールアドレスにアクセス用のアドレスとポートが送られてくる

vLabs Support

Update about Direct Access to the resources in your vLab "CEM Fabric Management and BMS Routing" Details

[redacted]

Hello [redacted]

ALLOWED NETWORK ADDRESS PREFIXES

Your IP address — [redacted] — now has 'direct access' to your sandbox.

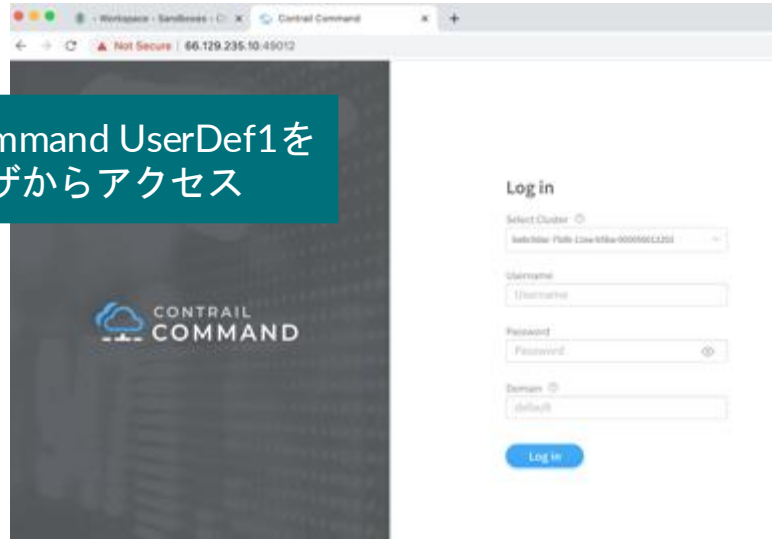
For more information on using 'direct access', check out the vLabs User Guide or the vLabs FAQ.

PORT FORWARDING INFORMATION

With your 'direct access' permission, you can now access the devices directly, as follows:

Abstract Resource Name	Protocol	Public IP Address	Public Port
Contrail_AIO	SSH	66.129.235.10	49001
Contrail_AIO	UserDef1	66.129.235.10	49002
Contrail_AIO	UserDef3	66.129.235.10	49004
Contrail_Appformix	SSH	66.129.235.10	49006
Contrail_Appformix	UserDef1	66.129.235.10	49007
Contrail_Appformix	UserDef3	66.129.235.10	49009
Contrail_Command	SSH	66.129.235.10	49011
Contrail_Command	UserDef1	66.129.235.10	49012
Contrail_Command	UserDef3	66.129.235.10	49014

Contrail Command UserDef1をブラウザからアクセス




```
Macbook:~ mAsaya$ sudo ssh jcluser@66.129.235.10 -p 49025
jcluser@66.129.235.10's password:
Last login: Wed Aug 19 19:30:58 2020 from kd059138041015.ppp-bb.dion.ne.jp
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory
[jcluser@CentOS ~]$ su
Password:
[root@CentOS jcluser]# virsh list
setlocale: No such file or directory
 Id    Name                           State
----------------------------------------------------
 1     vqfx-re                        running
 2     vqfx-pfe                       running

[root@CentOS jcluser]# virsh console 1
setlocale: No such file or directory
Connected to domain vqfx-re
Escape character is ^]

Password:
Login incorrect
login: jcluser
Password:

--- JUNOS 18.1R3-S5.3 built 2019-04-27 13:13:59 UTC
{master:0}
jcluser@Leaf1>
```

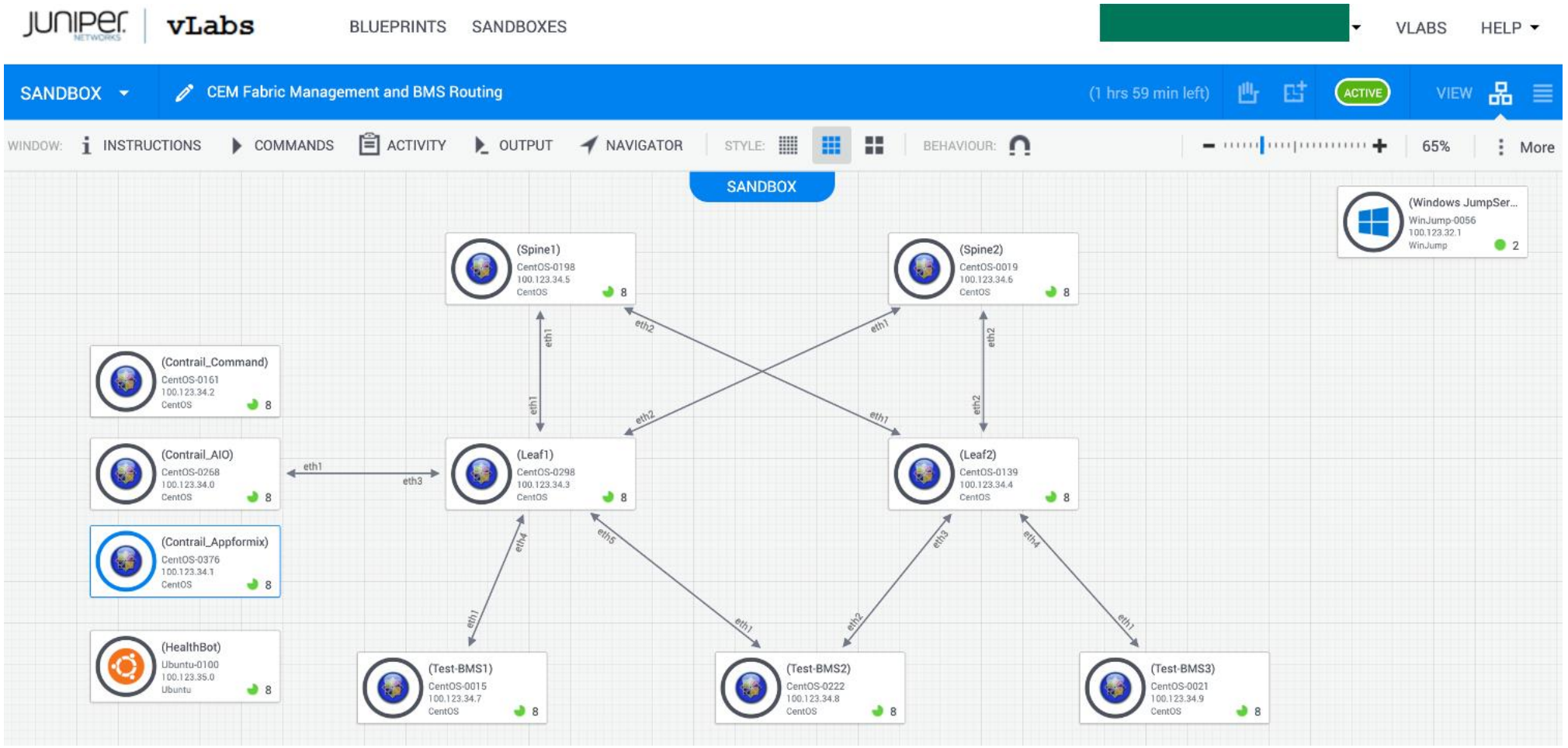
Leaf1のSSHアクセスを

参考：ブループリント利用手順

- Contrail Fabric Management and BMS Routing -

クラウド上の最新版はバージョン等変わっている可能性がございますので参考として参照ください。

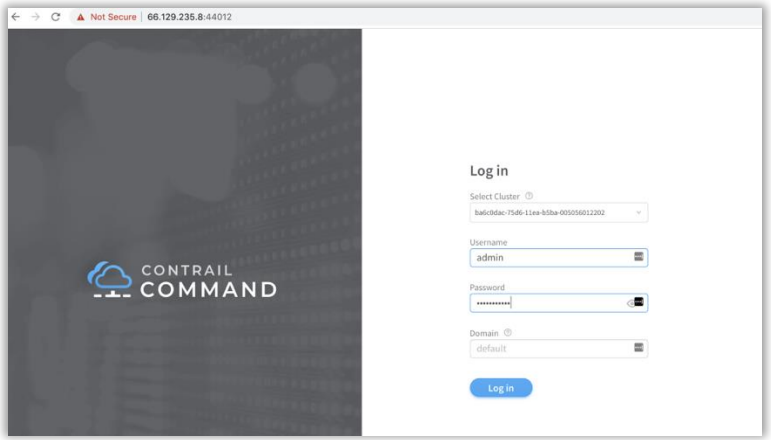
Contrail Fabric Management and BMS Routing



各種ソフトウェア GUIへのログイン

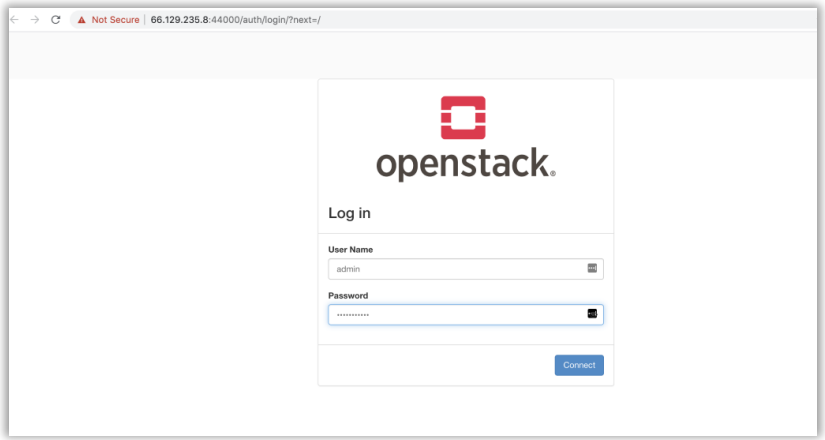
アクセス方法
※各デバイスへのアクセス方法が記載されたメールが通知されている

[Contrail Command]　Private Portが9091のもの
https://66.129.235.8:44012/



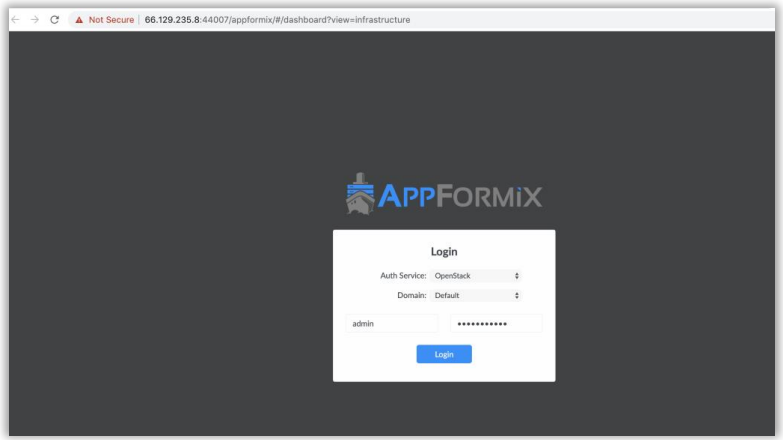
UserName: admin
Password: contrail123

[AIO(OpenStack)]
http://66.129.235.8:44000/



UserName: admin
Password: contrail123

[Appformix]
http://66.129.235.8:44007/



UserName: admin
Password: contrail123

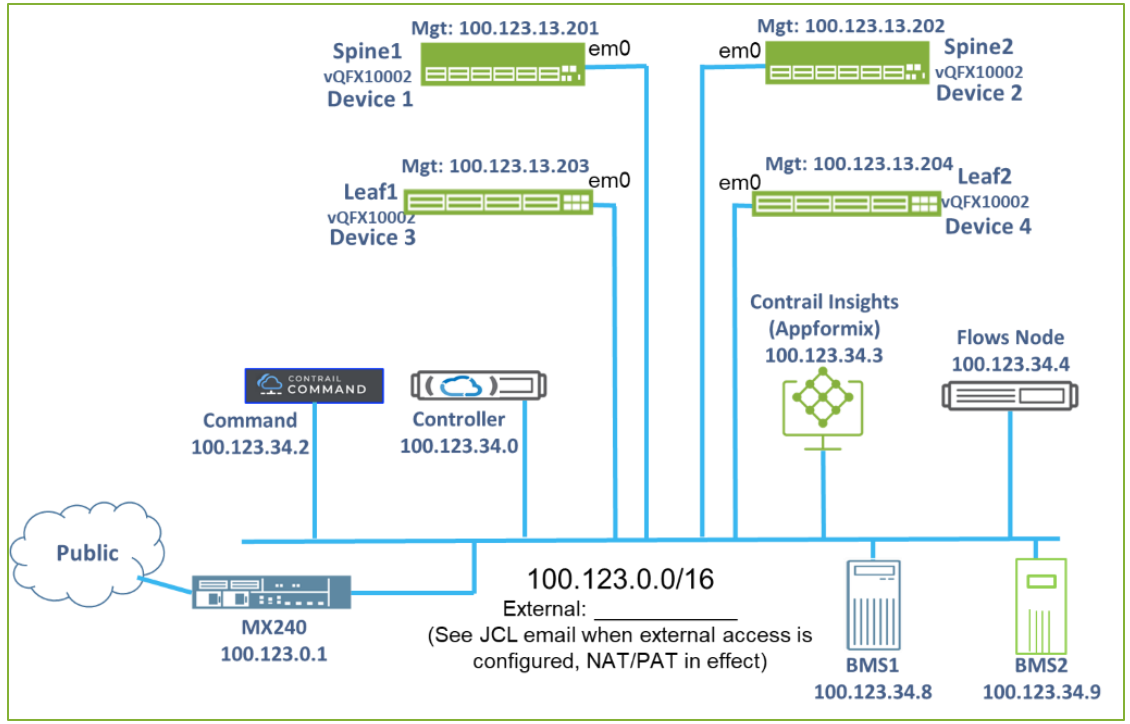
JUNOSデバイスへのログイン

[vQFX]
Leaf1 Sample:

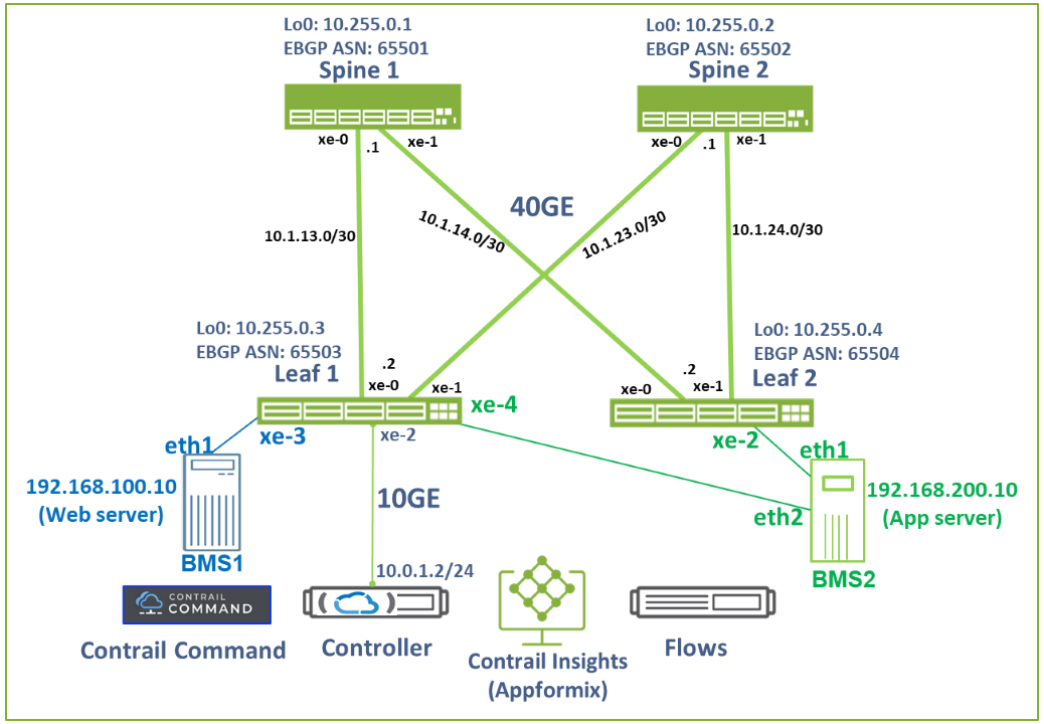
```
$ ssh root@66.129.235.8 -p 44024
root@66.129.235.8's password: Juniper!1

[root@CentOS ~]# virsh list
setlocale: No such file or directory
Id    Name                           State
----------------------------------------------------
1     vqfx-re                        running
2     vqfx-pfe                        running

[root@CentOS ~]# virsh console vqfx-re
Leaf1 (ttyd0)
login: root
Password: Juniper!1
root@Leaf1:RE:0% cli
{master:0}
root@Leaf1> show version
fpc0:
--------------------------------------------------------------------------
Hostname: Leaf1
Model: vqfx-10000
```

Topology

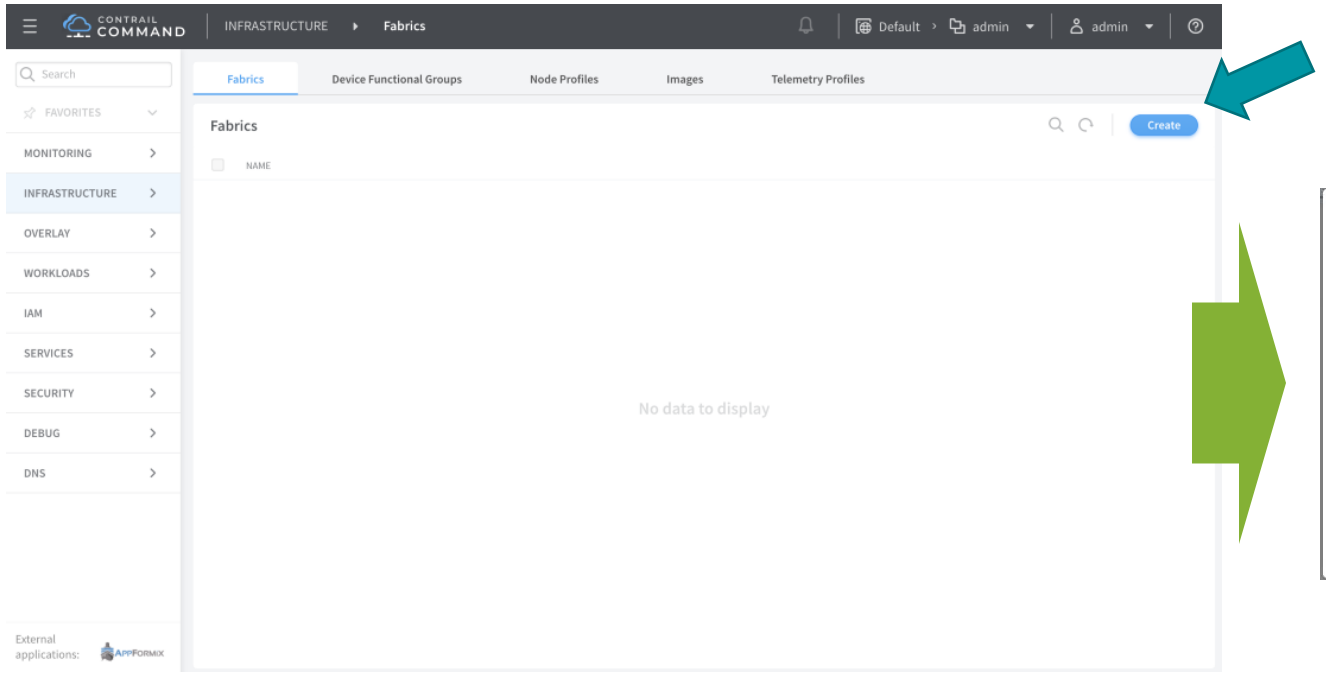
Management Topology



Dataplane Topology



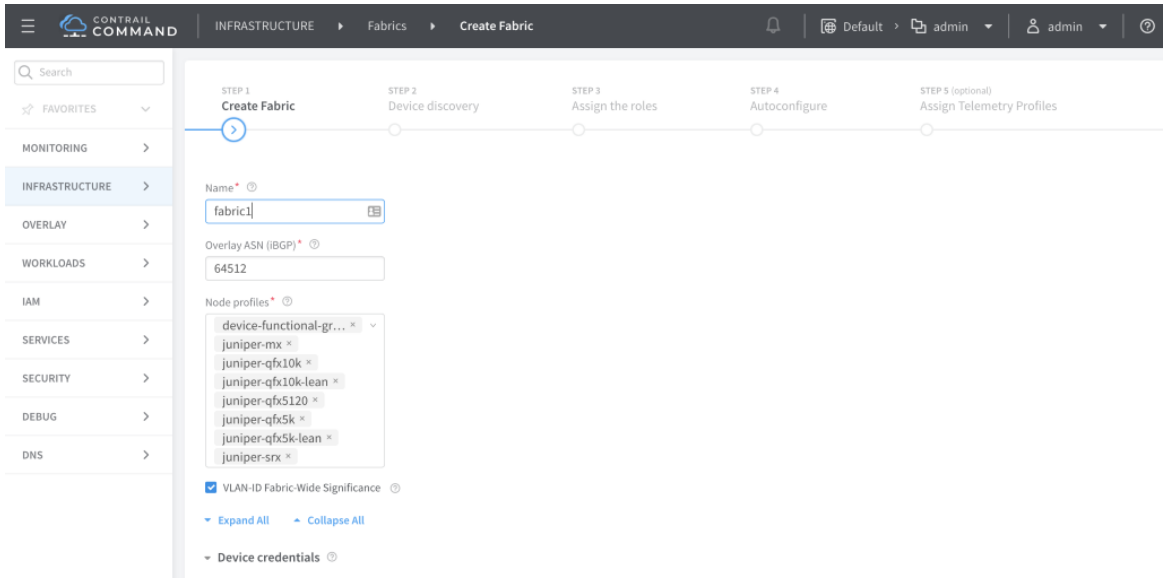
Demo1 – Fabric Onboarding (Brownfield)

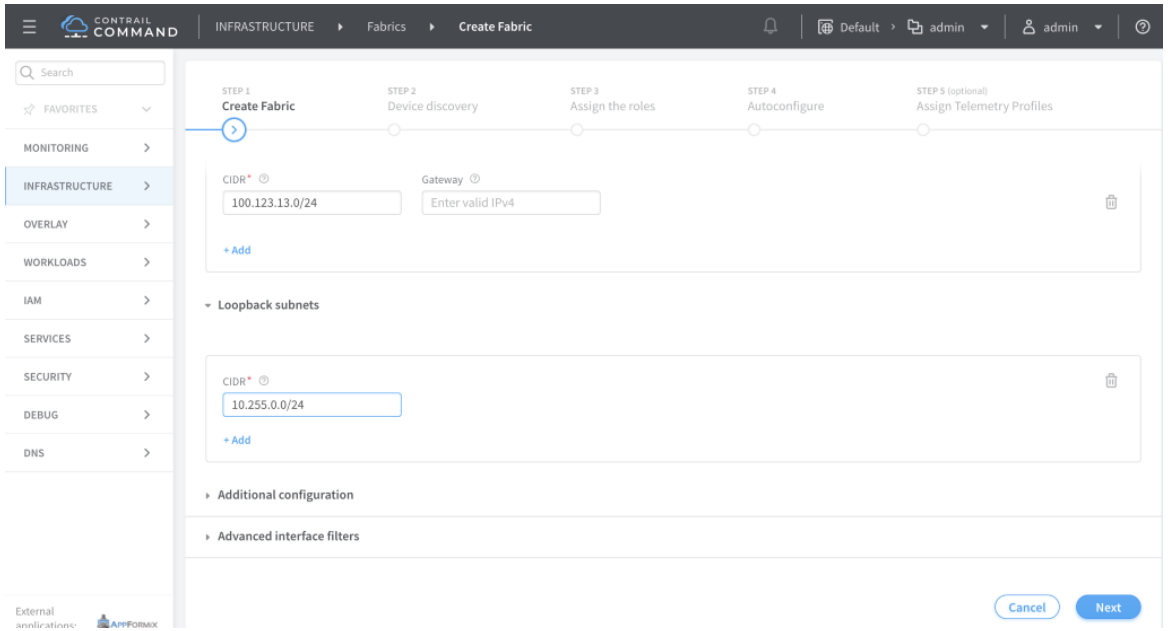


Existing Fabric(BrownFieldDeployment)を選択し、Provision

Demo1 – Fabric Onboarding (Brownfield)

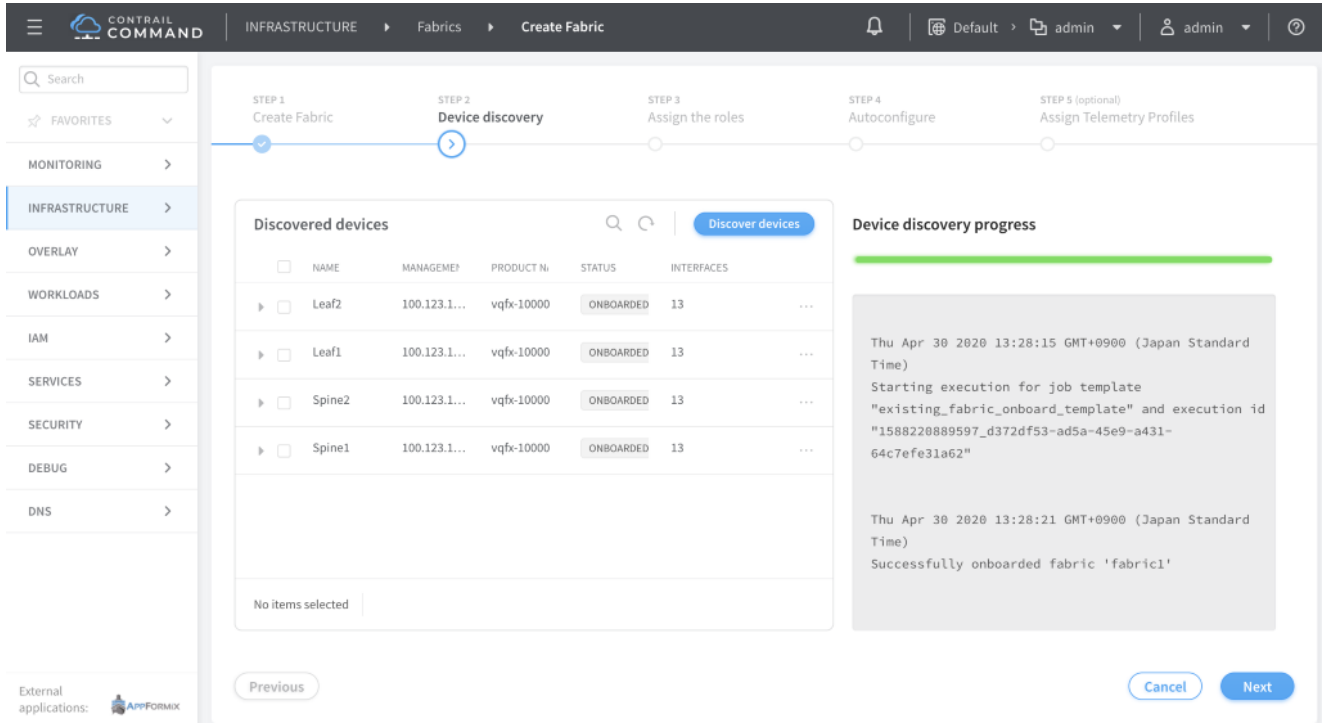
以下を入力
Name: 任意
Device credentials: root / Juniper!1
Management subnets: 100.123.13.0/24, Gateway:none
Loopback subnets: 10.255.0.0/24





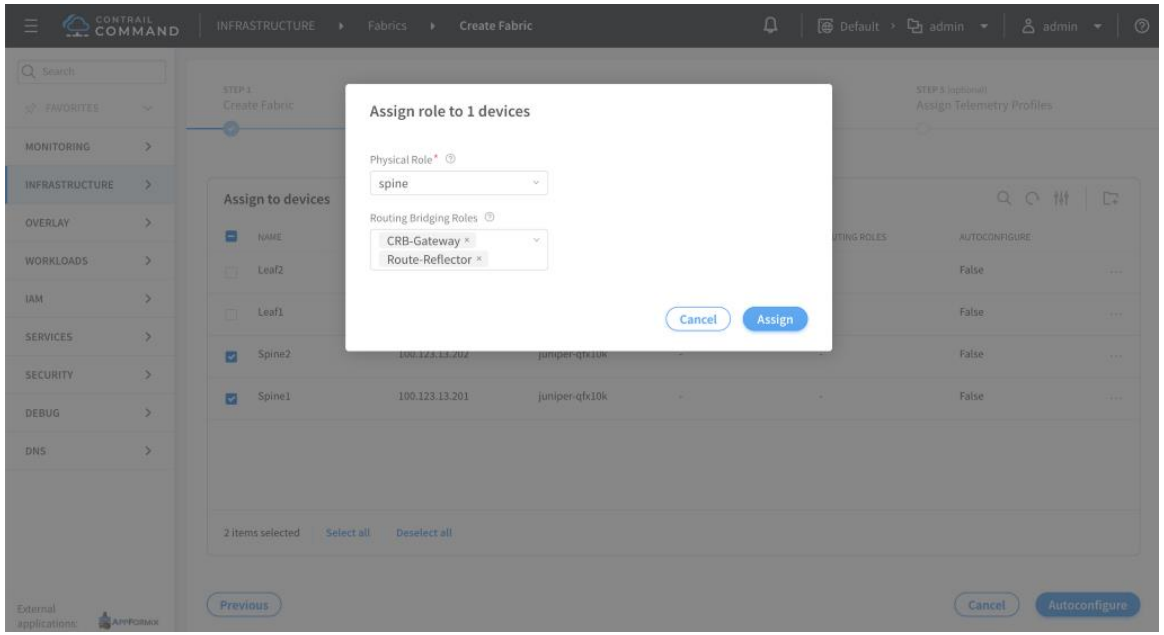
Demo1 – Fabric Onboarding (Brownfield)

Deviceが検出された後、Next

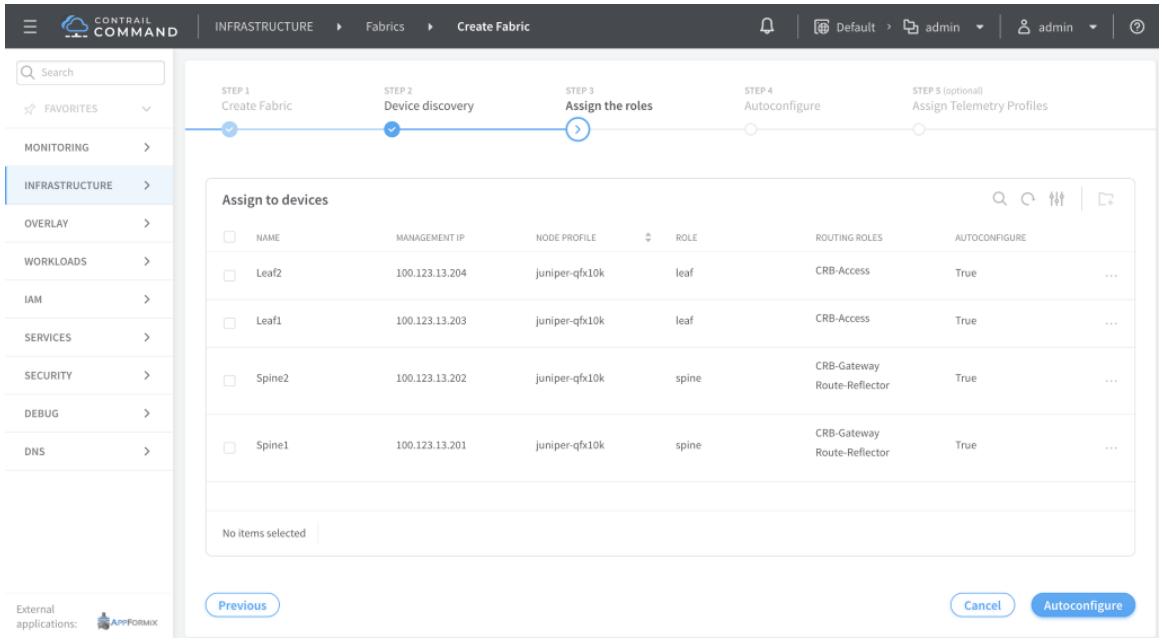


Demo1 – Fabric Onboarding (Brownfield)

各DeviceのRole設定
spine1,2 : PhysicalRole: spine RoutingBridgingRole: CRB-Gateway, Route-Reflector
leaf1,2 : PhysicalRole: leaf RoutingBridgingRole: CRB-Access

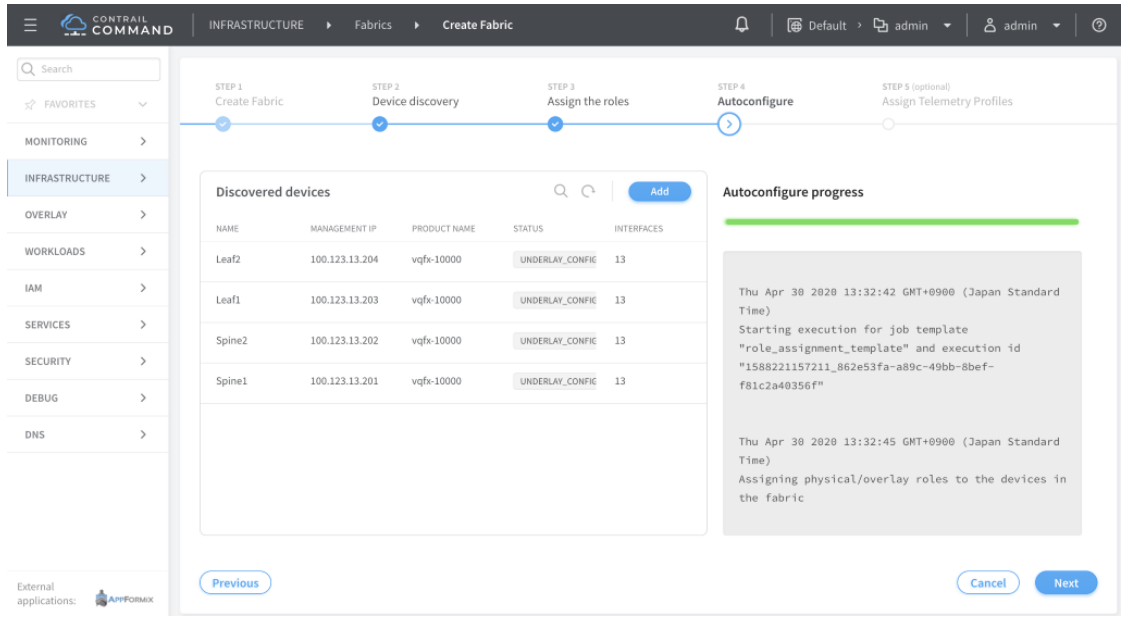


全てのDeviceのRole設定完了後、Autoconfigure

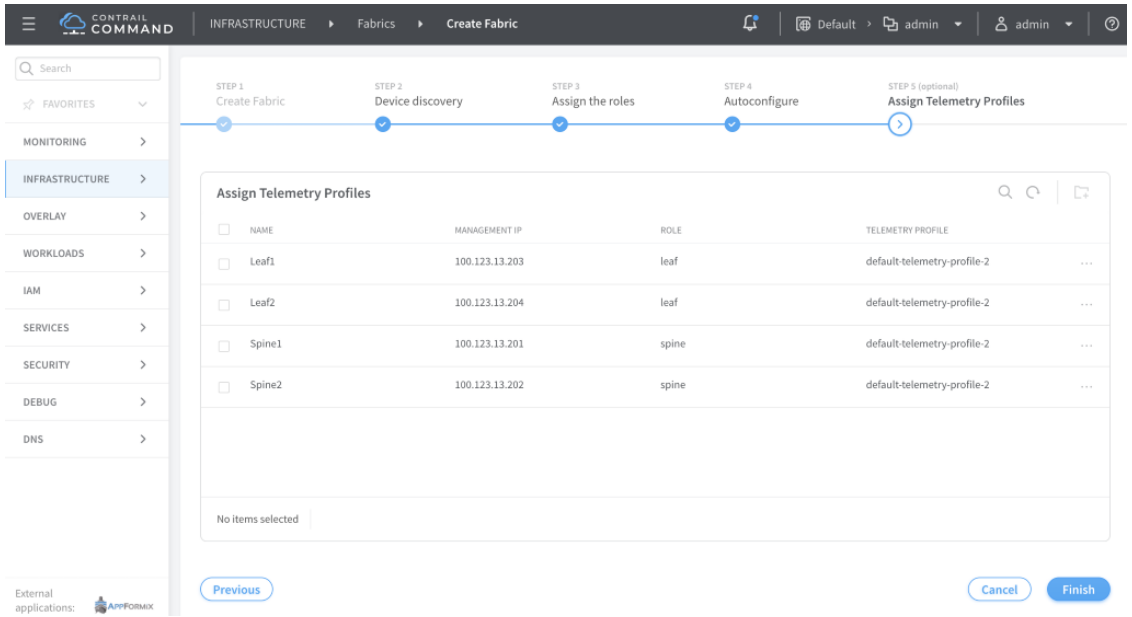


Demo1 – Fabric Onboarding (Brownfield)

Autoconfiguration processが完了後、 Next

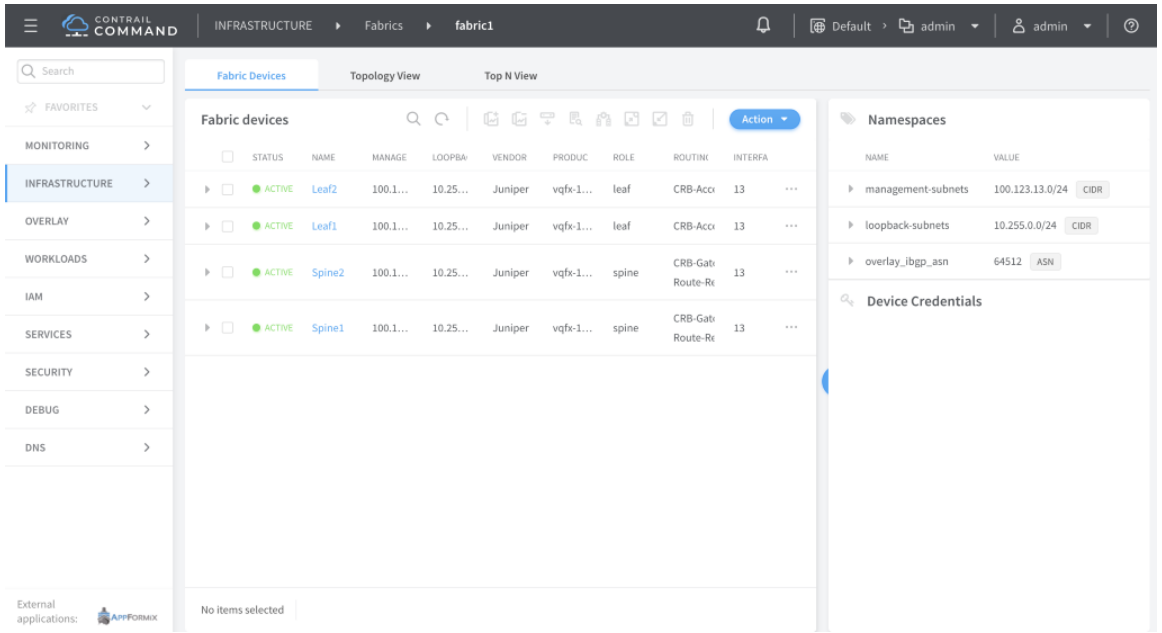


各Deviceにdefault-telemetry-profile2(Fabric Interfaceのみ有効)をアサイン
全てのDeviceに設定完了後、Finish

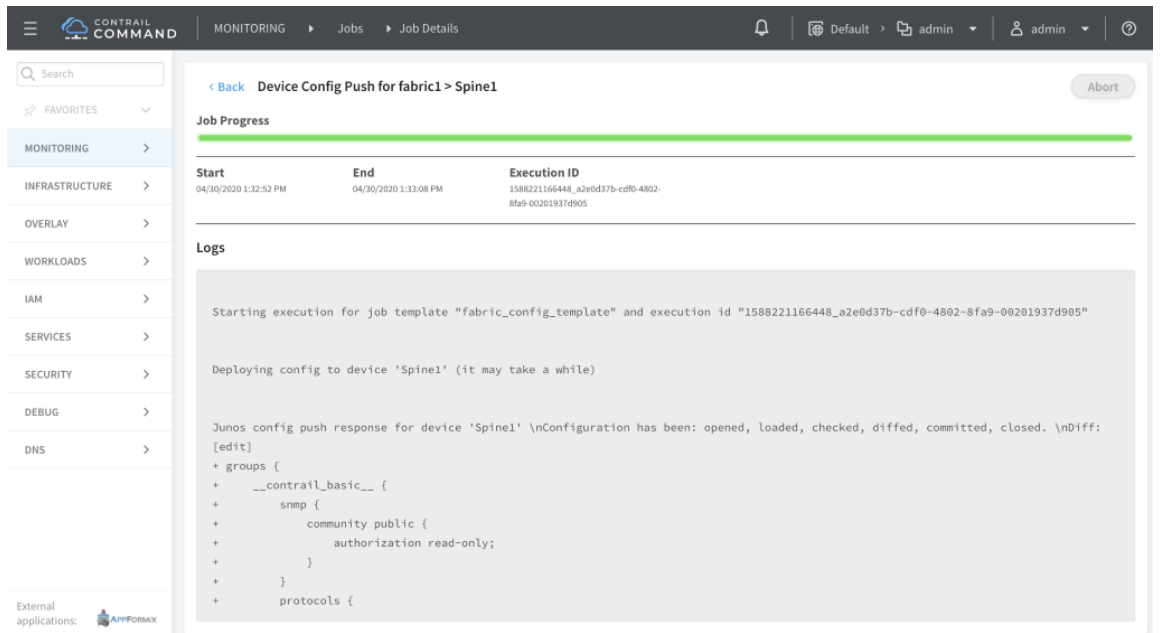


Demo1 – Fabric Onboarding (Brownfield)

設定完了を以下の画面でFabricの状態を確認可能



Job Typeを選択することで、Pushされたconfigを確認可能

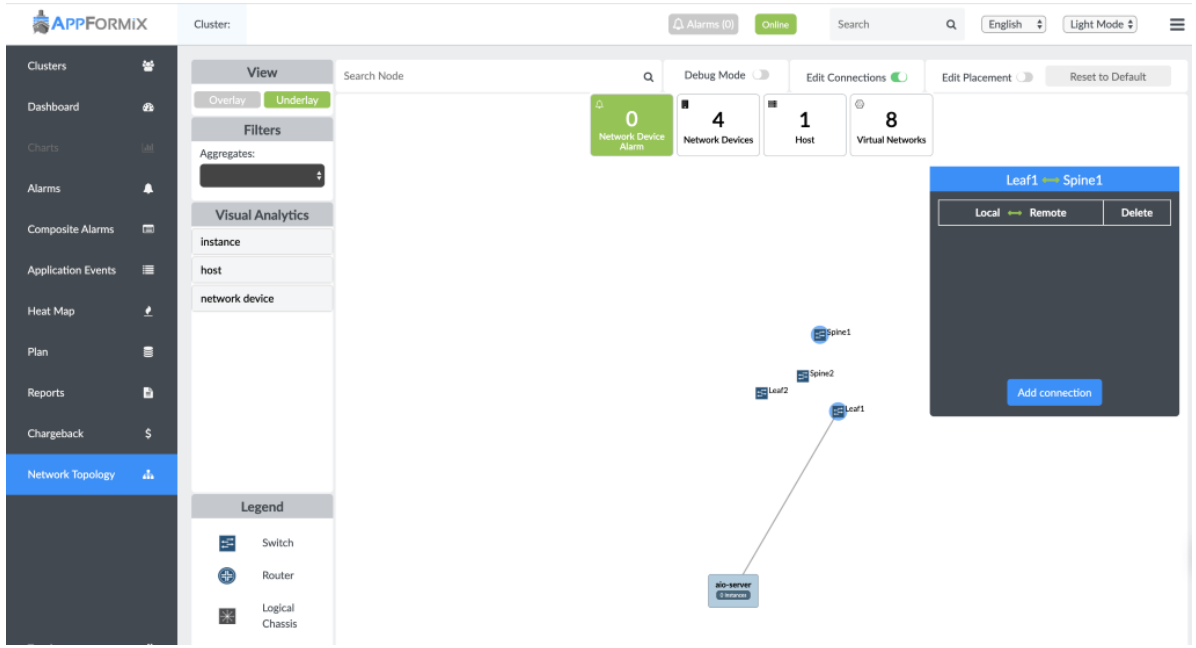


Demo1 – Fabric Onboarding (Brownfield)

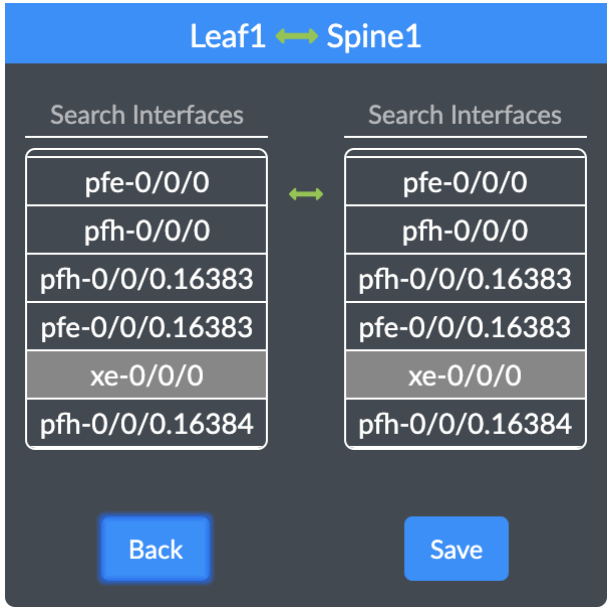
Topology View設定1 (NetworkConnection)

※vLABはdevice間でLLDPが通らないため本設定が必要)
ToplogyViewを表示せるために、Appformixにて以下の設定を実施

Appformixにloginし、"Network Topology"->"Edit Connections"をクリック
Spine1とLeaf1を選択し、Add connectionをクリック



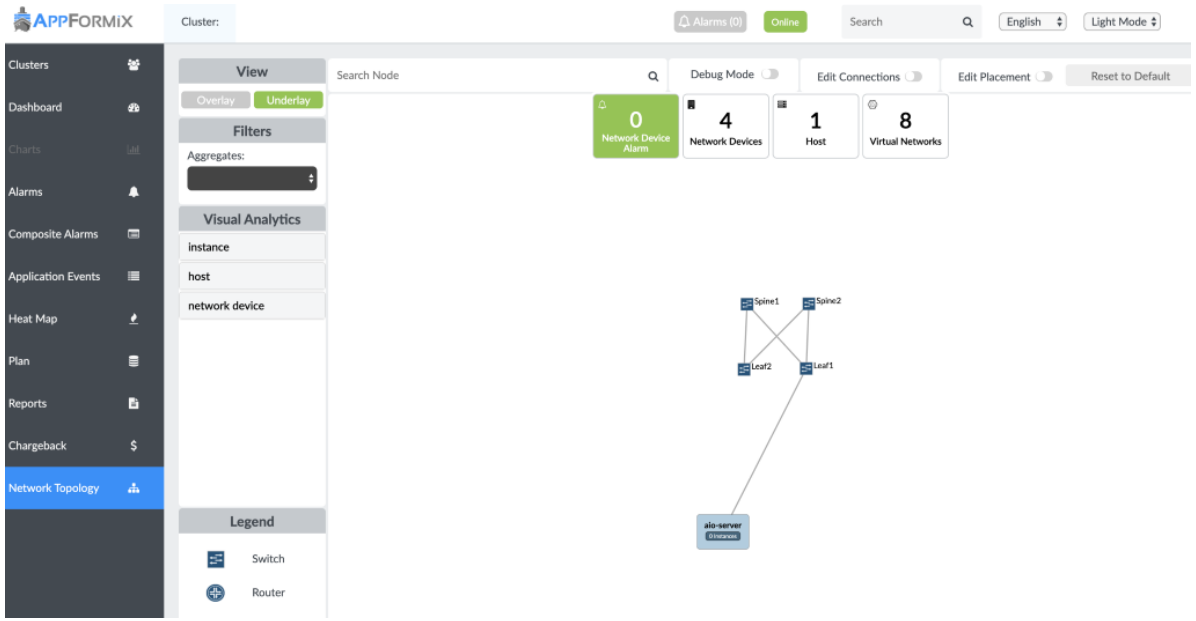
Leaf1側xe-0/0/0, Spine1側xe-0/0/0を選択し、Save



上記を繰り返し、以下の接続になるように設定
spine1(xe-0/0/0) - leaf1(xe-0/0/0)
spine1(xe-0/0/1) - leaf2(xe-0/0/0)
spine2(xe-0/0/0) - leaf1(xe-0/0/1)
spine2(xe-0/0/1) - leaf2(xe-0/0/1)

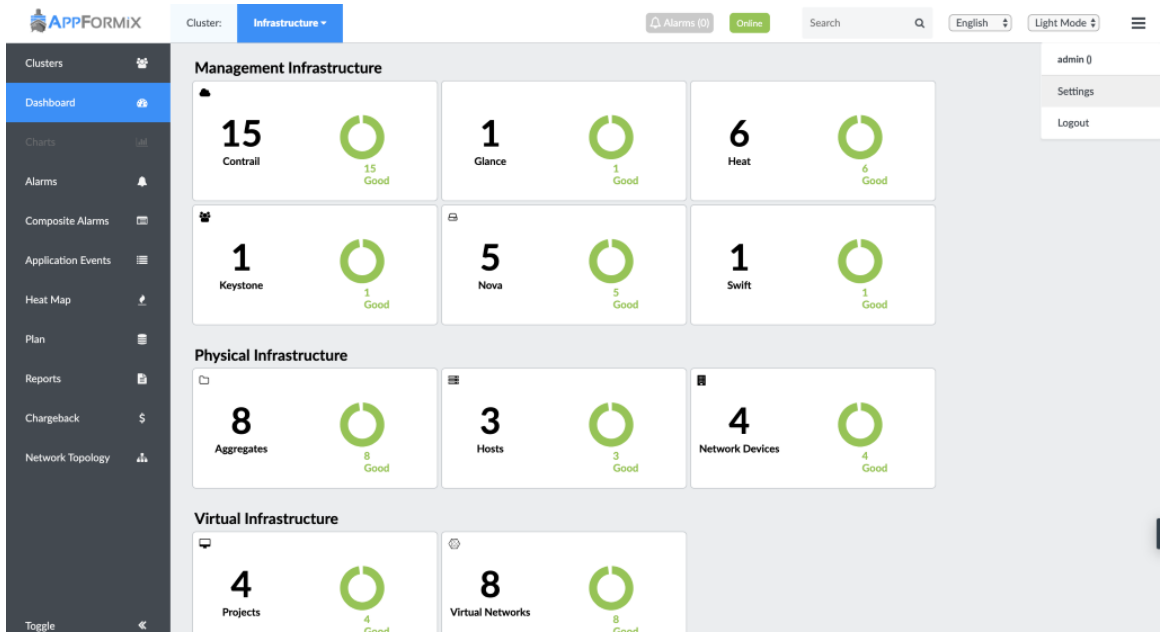
Demo1 – Fabric Onboarding (Brownfield)

設定完了後、以下のようにTopologyが表示されることを確認

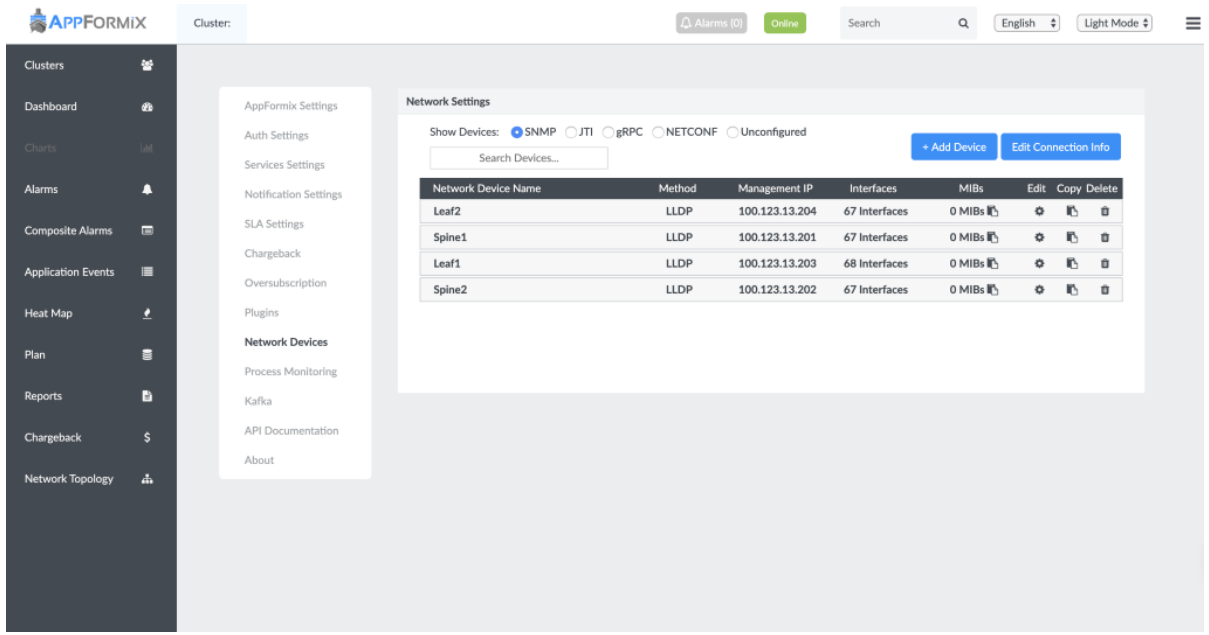


Demo1 – Fabric Onboarding (Brownfield)

Topology View設定2 (SNMP設定)
AppformixにLoginし、"Setting"を選択

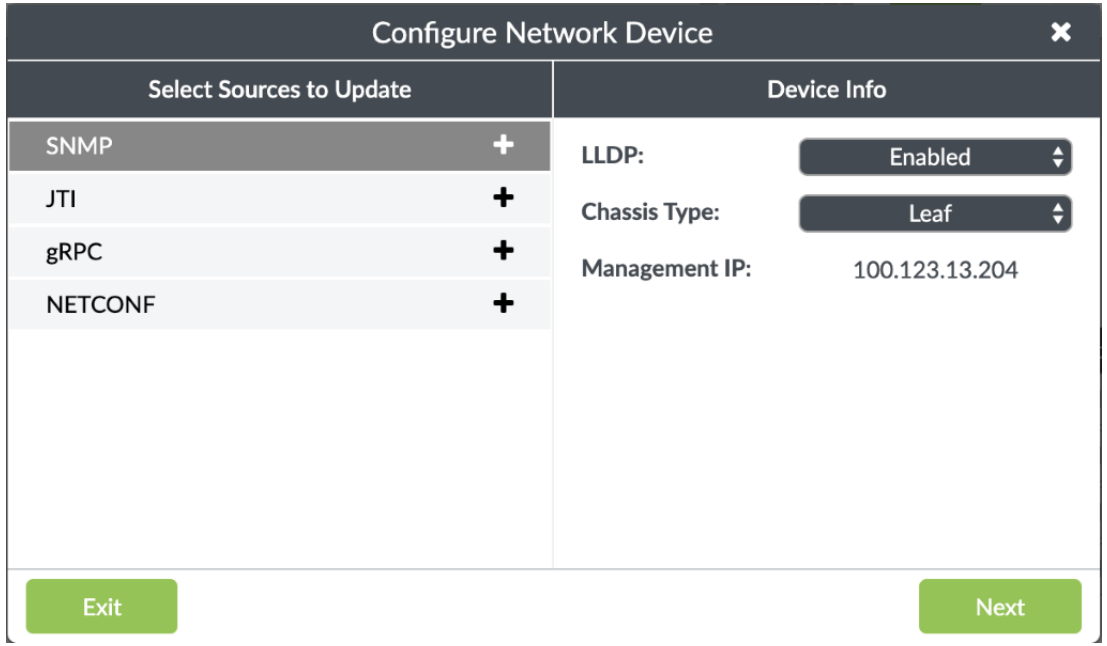


"NetworkDevice"にて、ContrailCommandでFabricを作成した各Deviceが表示されていることを確認し、"Edit"をクリック

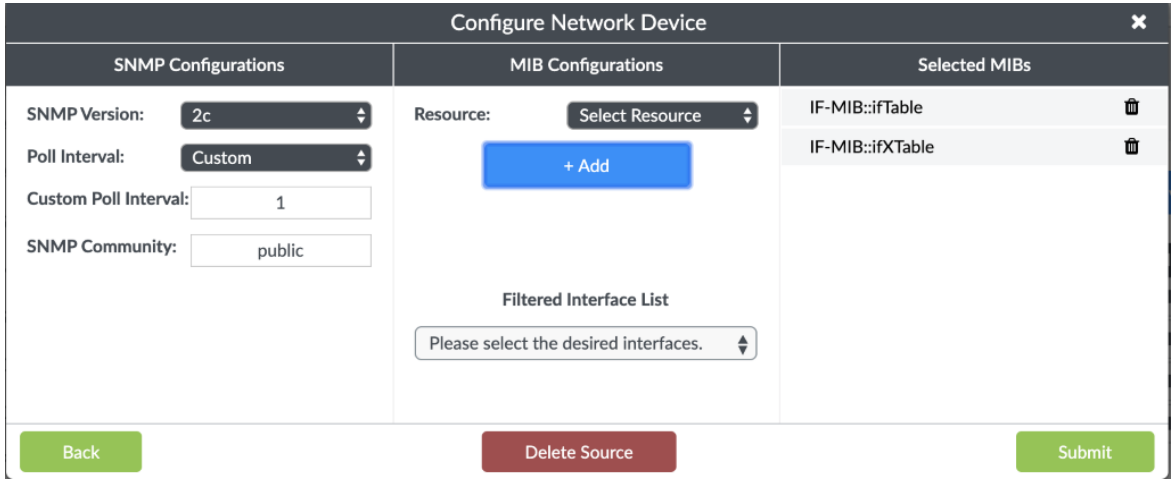


Demo1 – Fabric Onboarding (Brownfield)

SNMPを選択し、Next

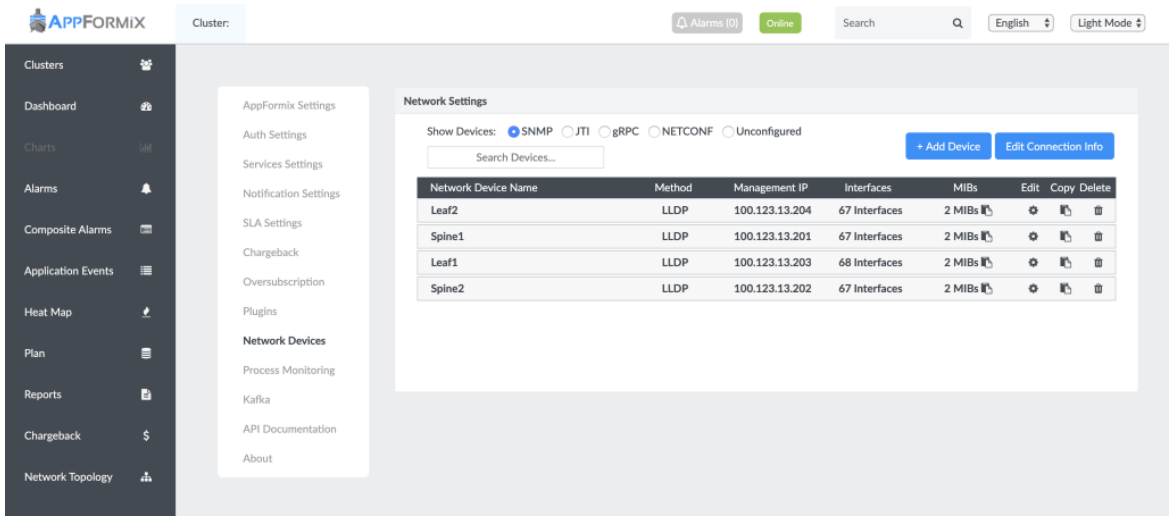


"MIB Configurations"にて以下のMIBを選択し、Add
IF-MIB::ifTable
IF-MIB::ifXTable
Submitをクリック

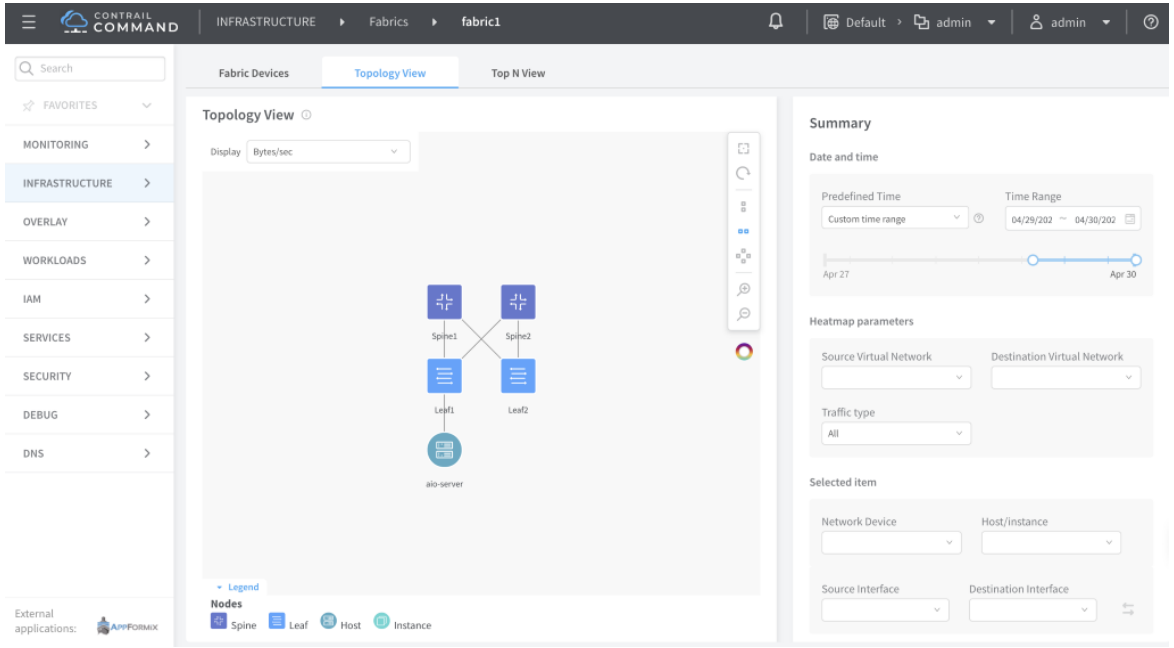


Demo1 – Fabric Onboarding (Brownfield)

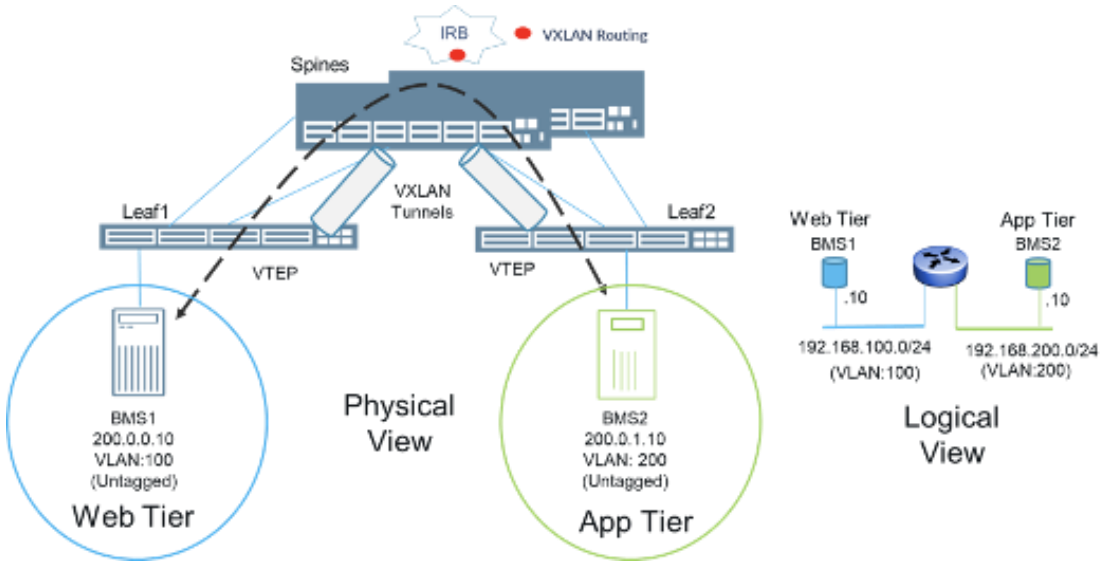
各Deviceに同MIBを設定後、以下のように表示されていることを確認



Contrail CommandにLoginし、TopologyViewが表示されることを確認



Demo2 – BMS間通信



BMS1, BMS2にLoginし、IP,Routeを確認

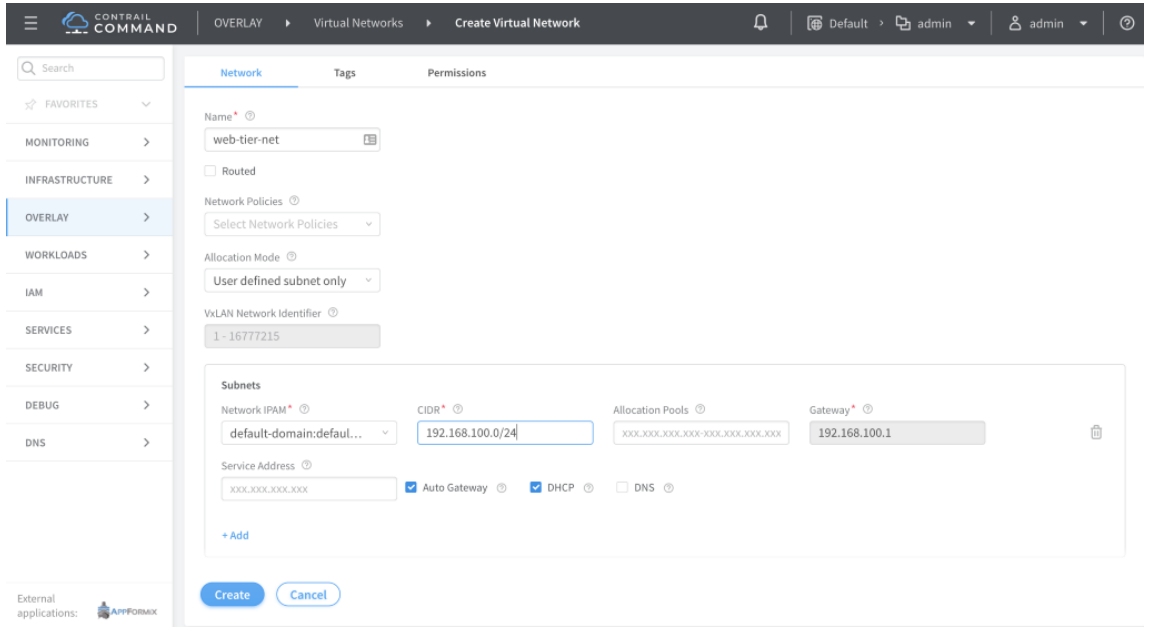
```
[root@bms1 ~]# ifconfig eth1
eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.100.10  netmask 255.255.255.0  broadcast 192.168.100.255

[root@bms1 ~]# ip route
default via 100.123.0.1 dev eth0 proto static metric 100
100.123.0.0/16 dev eth0 proto kernel scope link src 100.123.34.8 metric 100
192.168.100.0/24 dev eth1 proto kernel scope link src 192.168.100.10 metric 100
192.168.200.0/24 via 192.168.100.1 dev eth1 proto static metric 100
```

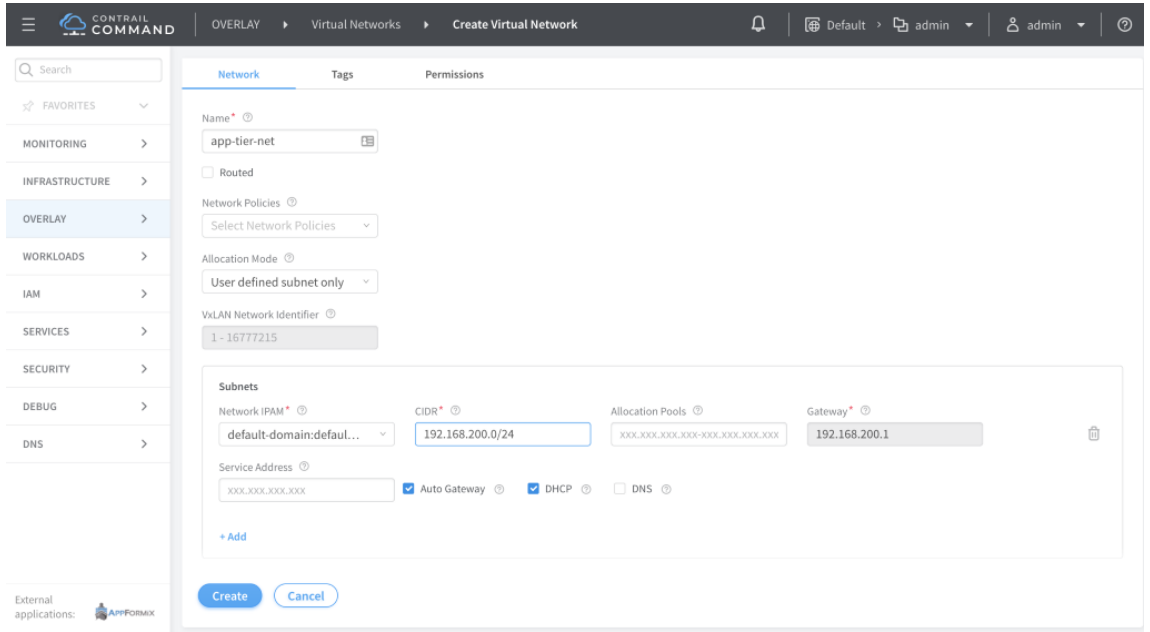
```
[root@bms2 ~]# ifconfig eth2
eth2: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.200.10  netmask 255.255.255.0  broadcast 192.168.200.255

[root@bms2 ~]# ip route
default via 100.123.0.1 dev eth0 proto static metric 100
100.123.0.0/16 dev eth0 proto kernel scope link src 100.123.34.9 metric 100
169.254.0.0/16 dev eth2 scope link metric 1004
192.168.100.0/24 via 192.168.200.1 dev eth2
192.168.200.0/24 dev eth2 proto kernel scope link src 192.168.200.10
```

Demo2 – BMS間通信

VN作成

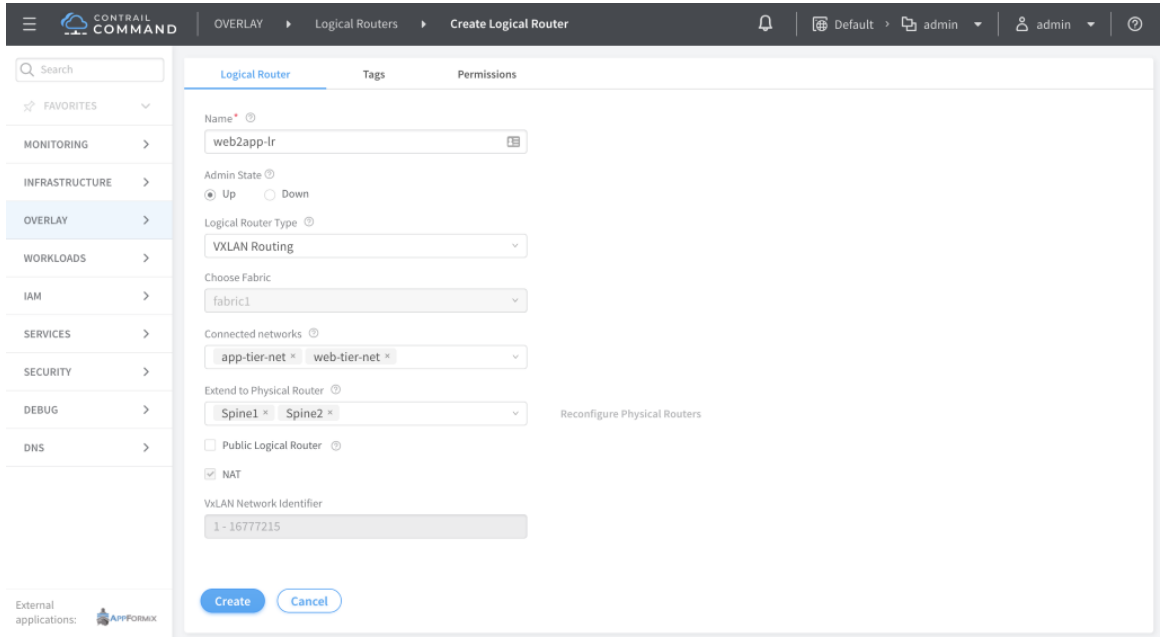
Overlay->VirtualNetworksにて、Createをクリックし、以下の２つのVNを作成
web-tier-net: 192.168.100.0/24 for BMS1
app-tier-net: 192.168.200.0/24 for BMS2





Demo2 – BMS間通信

LR作成
以下の通りLogicalRouterを作成
LogicalRouterType: VXLAN Routing
Fabric: fabric1
Connected networks: app-tier-net, web-tier-net
Extended to Physical Router: Spine1, Spine2



Monitoring->Jobから以下の設定がPushされていることが確認できる
Sample: spine1

```
edit groups __contrail_overlay_evpn__ protocols evpn]
+    vni-options {
+        vni 10 {
+            vrf-target target:64512:8000010;
+        }
+        vni 9 {
+            vrf-target target:64512:8000009;
+        }
+    }
[edit groups __contrail_overlay_evpn__ policy-options]
+    policy-statement __contrail_web2app-lr_1f4f1b6e-6921-4dfa-a17b-907033d9857b-import {
+        term t1 {
+            from community target_64512_8000012;
+            then accept;
+        }
+    }
+    policy-statement __contrail_web2app-lr_1f4f1b6e-6921-4dfa-a17b-907033d9857b-export {
+        term t1 {
+            then {
+                community add target_64512_8000012;
+                accept;
+            }
+        }
+    }
------
```

Demo2 – BMS間通信

Spine1, Spine2にLoginし、IRBが作成されていることを確認

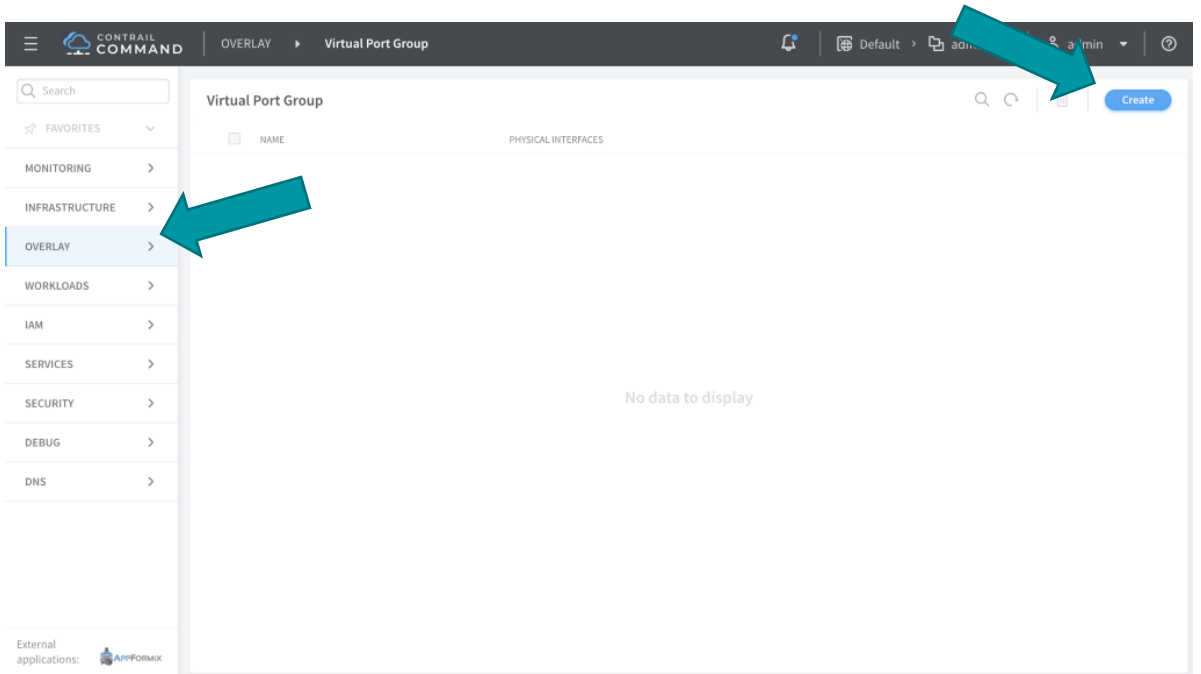
```
root@Spine1> show interfaces terse irb
Interface           Admin Link Proto    Local                 Remote
irb                 up    up
irb.9               up    up   inet     192.168.100.1/24
                                        192.168.100.4/24
irb.10              up    up   inet     192.168.200.1/24
                                        192.168.200.4/24

root@Spine2> show interfaces terse irb
Interface           Admin Link Proto    Local                 Remote
irb                 up    up
irb.9               up    up   inet     192.168.100.1/24
                                        192.168.100.5/24
irb.10              up    up   inet     192.168.200.1/24
                                        192.168.200.5/24
```

VPG作成

Overlay->VirtualPortGroupにて、以下の通りBMS1, BMS2用のVPGを作成

BMS1 - (vlan100 untag)(xe-0/0/3)leaf1

BMS2 - (vlan200 untag)(xe-0/0/2)leaf2



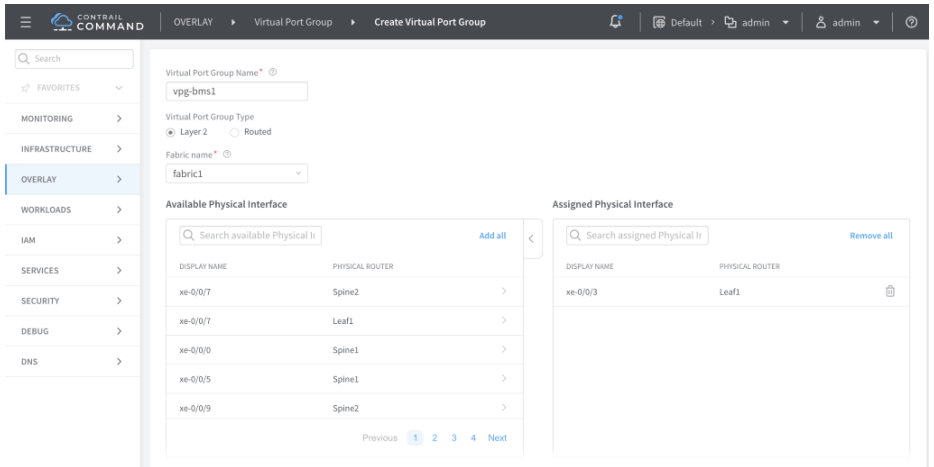
Demo2 – BMS間通信

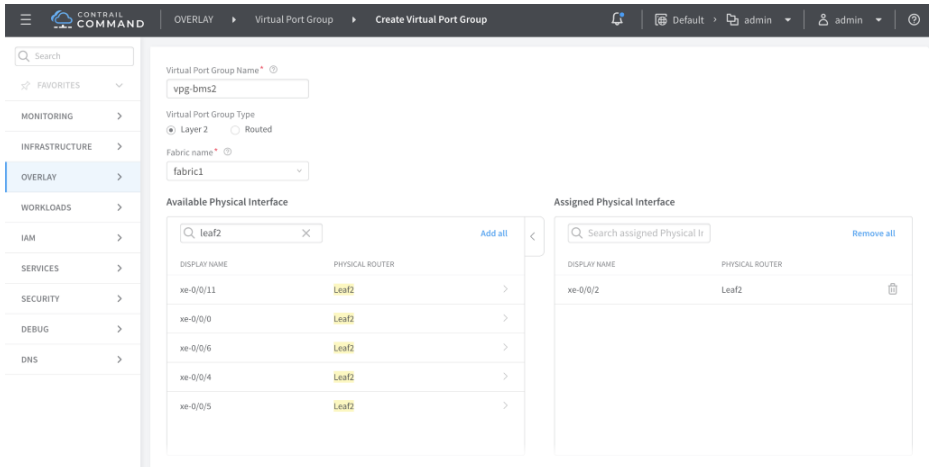
VPG作成

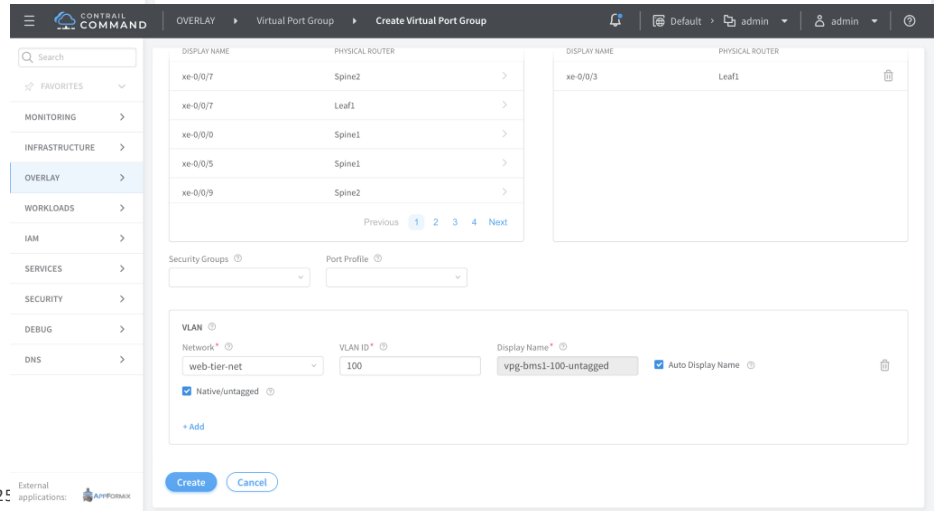
Overlay->VirtualPortGroupにて、以下の通りBMS1, BMS2用のVPGを作成

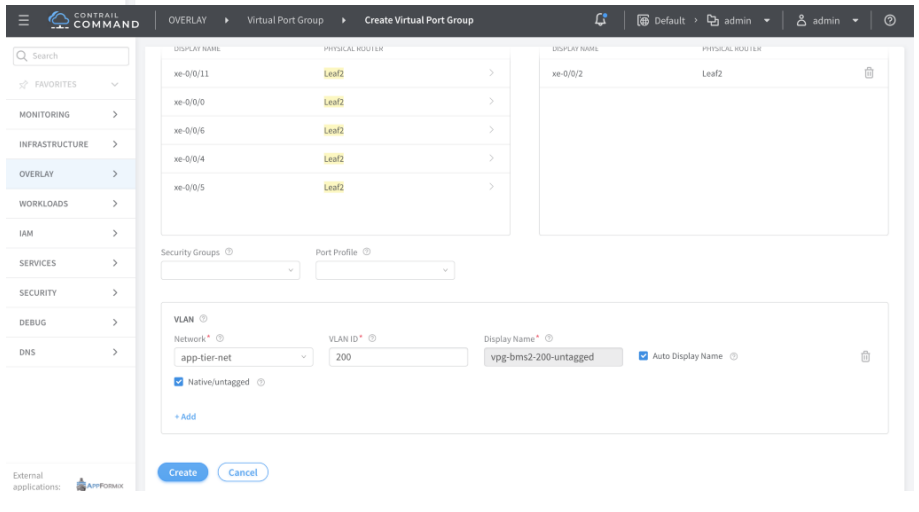
BMS1 - (vlan100 untag)(xe-0/0/3)leaf1

BMS2 - (vlan200 untag)(xe-0/0/2)leaf2









Demo2 – BMS間通信

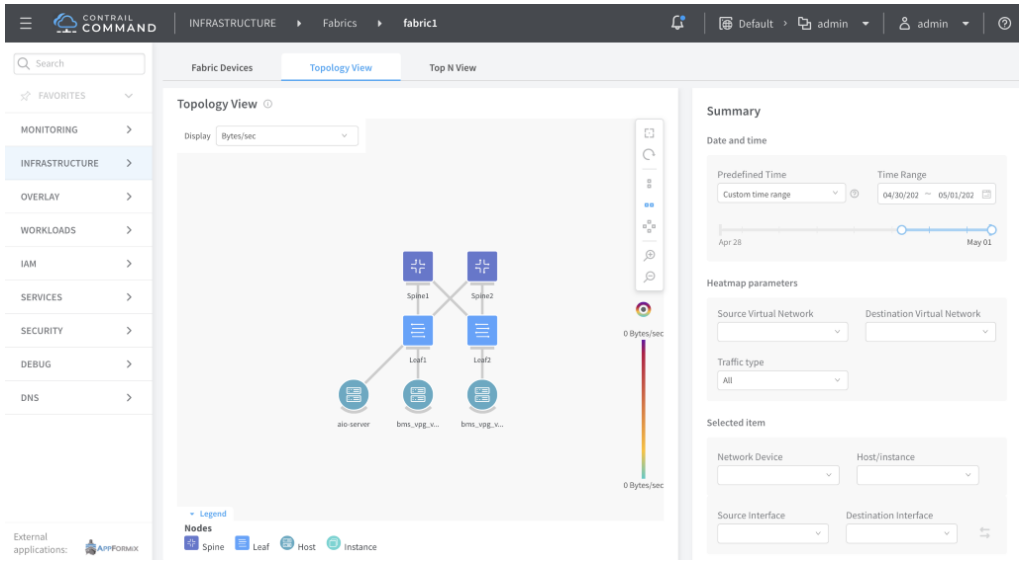
Monitoring->Jobから以下の設定がPushされていることが確認できる
Sample: leaf1

```
[edit groups __contrail_overlay_evpn__ protocols evpn]
+    vni-options {
+        vni 9 {
+            vrf-target target:64512:8000009;
+        }
+    }
[edit groups __contrail_overlay_evpn__ policy-options]
+   policy-statement _contrail_web-tier-net-l2-9-import {
+       term t1 {
+           from community target_64512_8000009;
+           then accept;
+       }
+   }
+ -----
Job summary: Job execution completed successfully.Detailed job results:
Successfully completed job for 1 devices.
```

BMS1にLoginし、BMS2との疎通確認

```
[root@bms1 ~]# ping 192.168.200.10
PING 192.168.200.10 (192.168.200.10) 56(84) bytes of data.
64 bytes from 192.168.200.10: icmp_seq=1 ttl=63 time=119 ms
64 bytes from 192.168.200.10: icmp_seq=2 ttl=63 time=205 ms
64 bytes from 192.168.200.10: icmp_seq=3 ttl=63 time=121 ms
```

Contrail CommandにてBMSの接続が確認可能
vLABではTelemetryがうまく動作しないため、Traffic流量の確認は不可



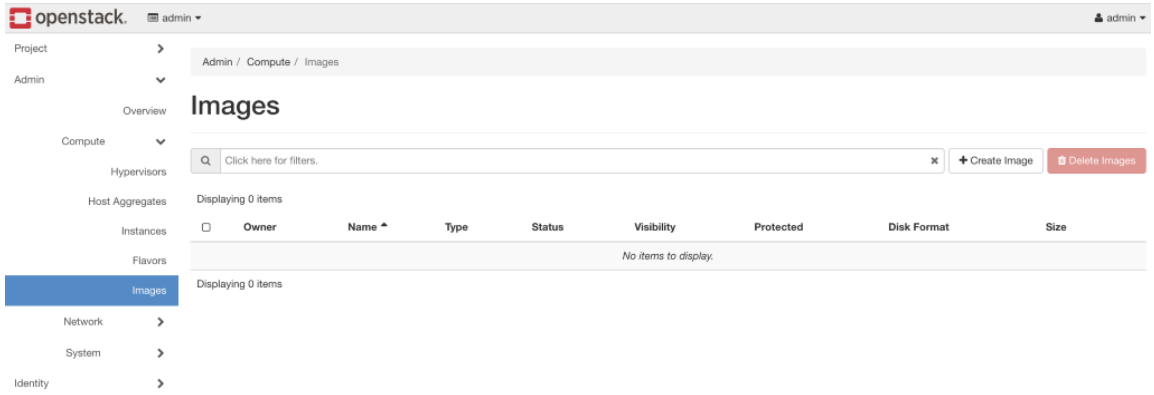
Demo3 – OpenStack VM, BMS間接続

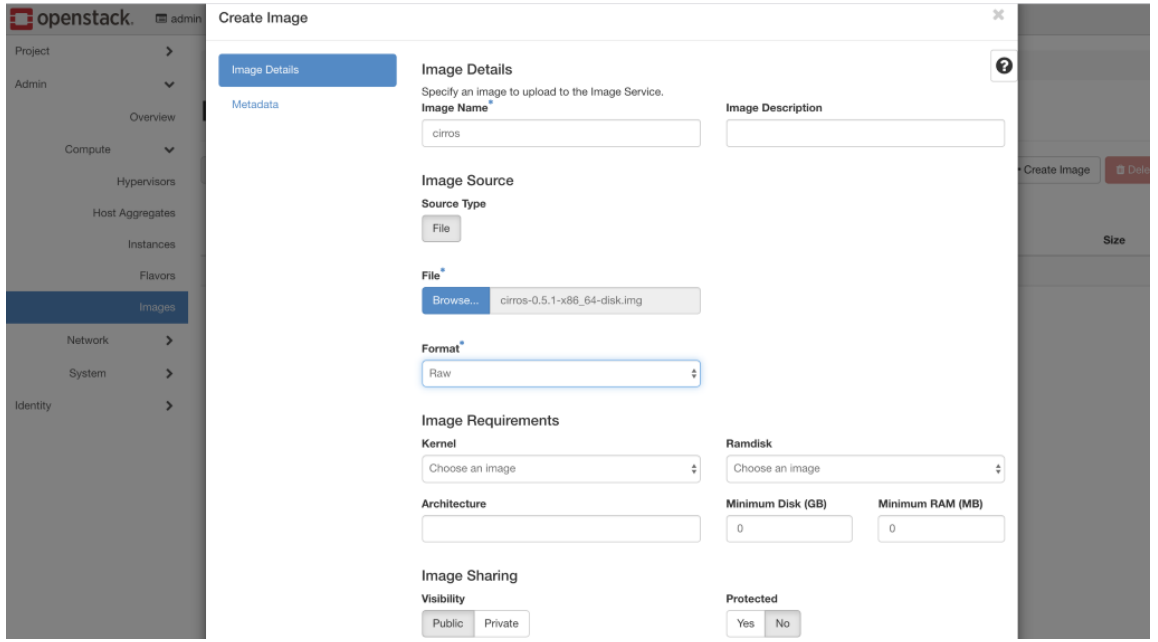
＊vLAB環境だと動作しない。セットアップオペレーションのみ

OpenStack準備

以下からCirrOSをDownload

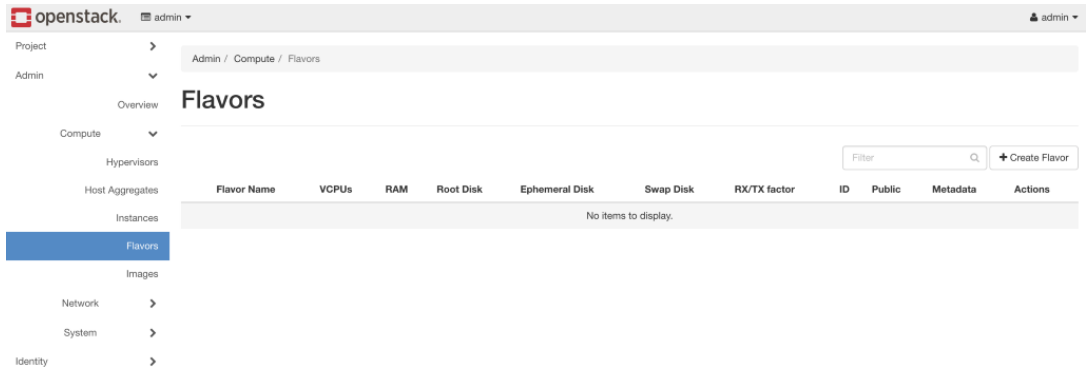
http://download.cirros-cloud.net/0.5.1/cirros-0.5.1-x86_64-disk.img

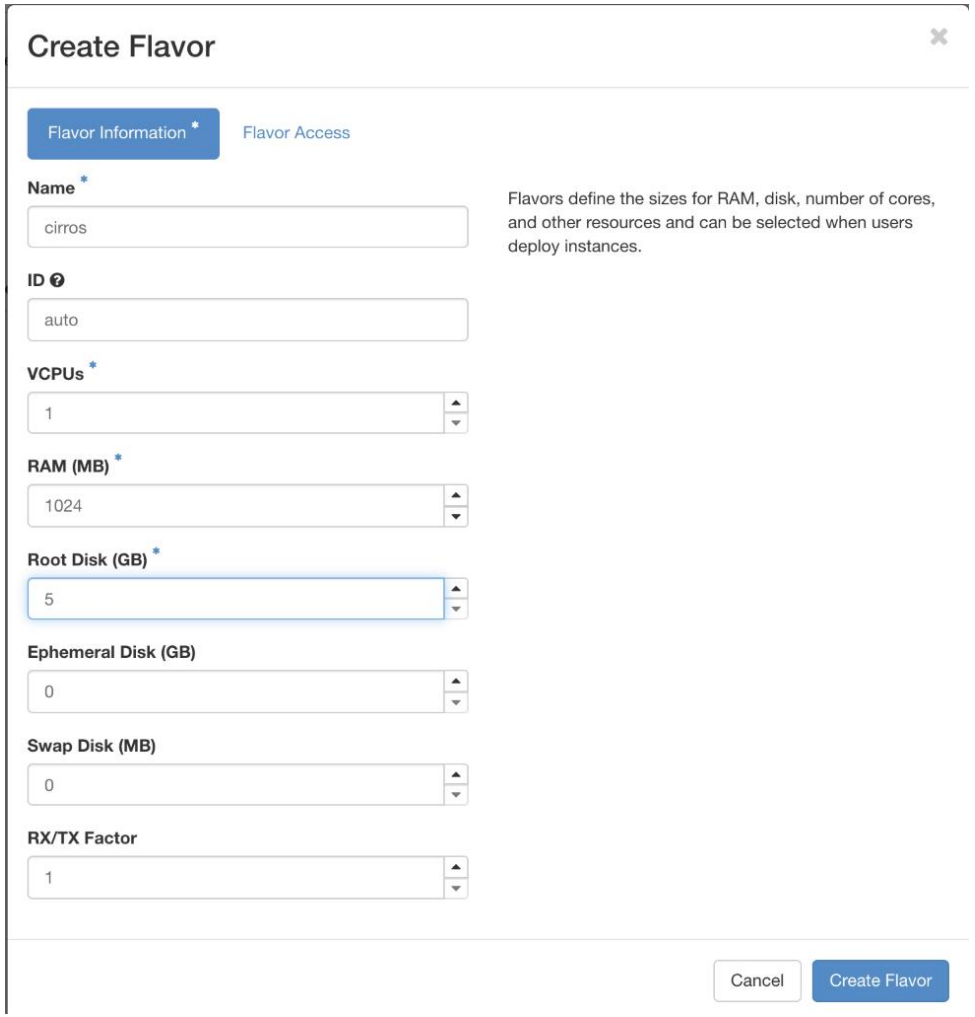




Demo3 – OpenStack VM, BMS間接続

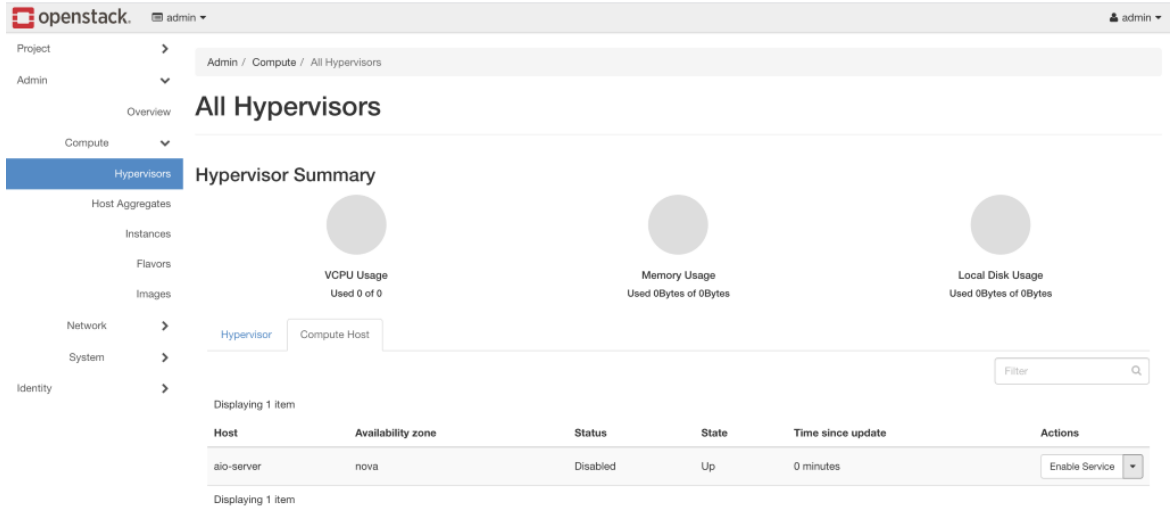
Flavor作成
Admin->Compute->Flavorsにて、以下の通りFlavorを作成



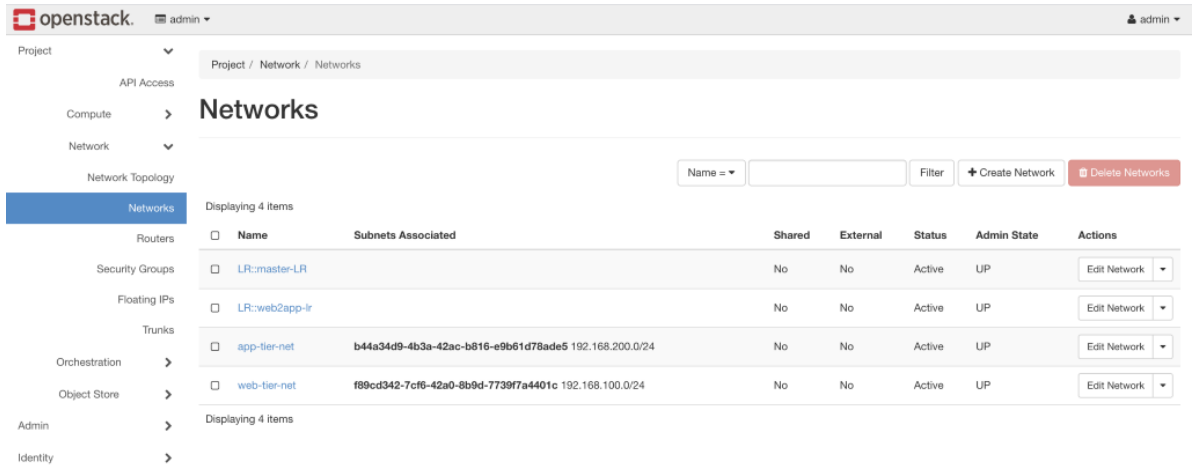


Demo3 – OpenStack VM, BMS間接続

Hypervisor有効化
Admin->Compute->Hypervisors->ComputeHostにてaio-serverをEnable

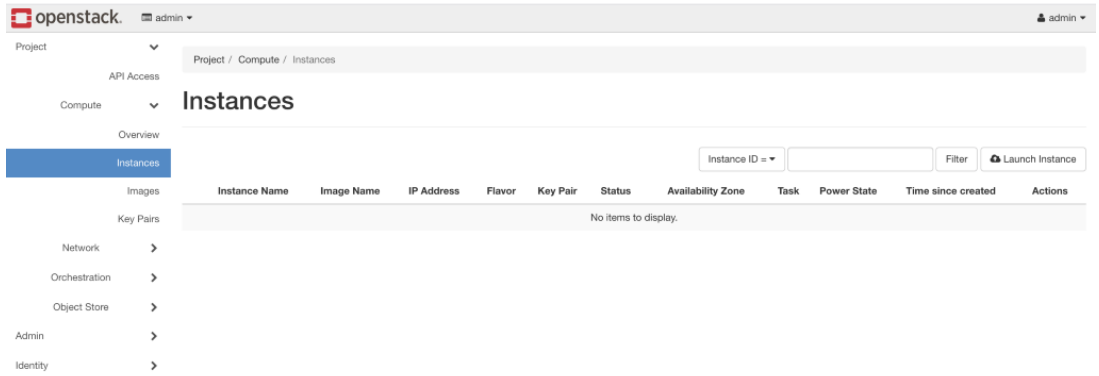


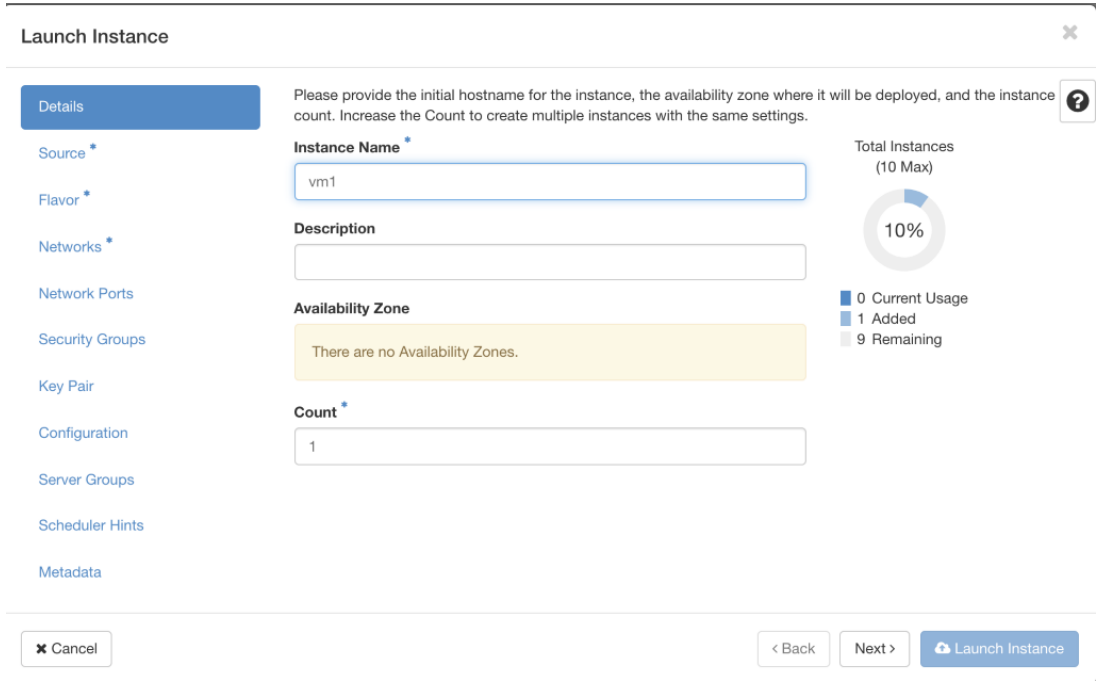
Network確認
Project->Network->Networksにて、Demo2で作成したSubnetが
OpenStack側にも反映されていることを確認



Demo3 – OpenStack VM, BMS間接続

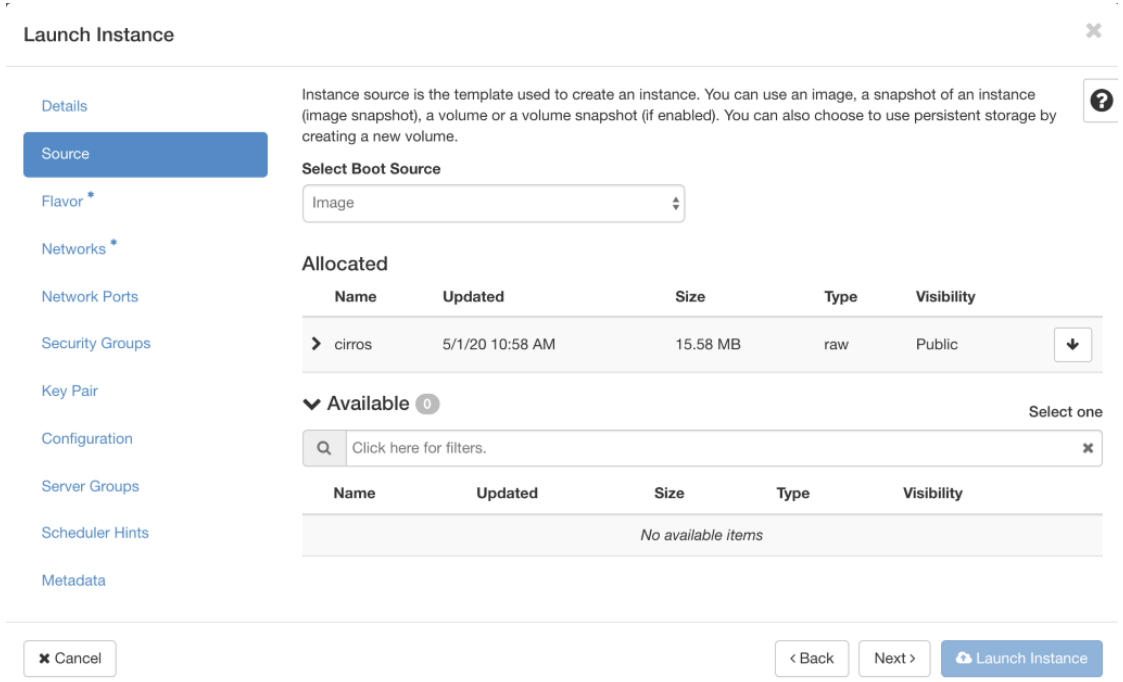
Instance作成
Project->Compute->Instancesにて、Launch Instancesをクリック



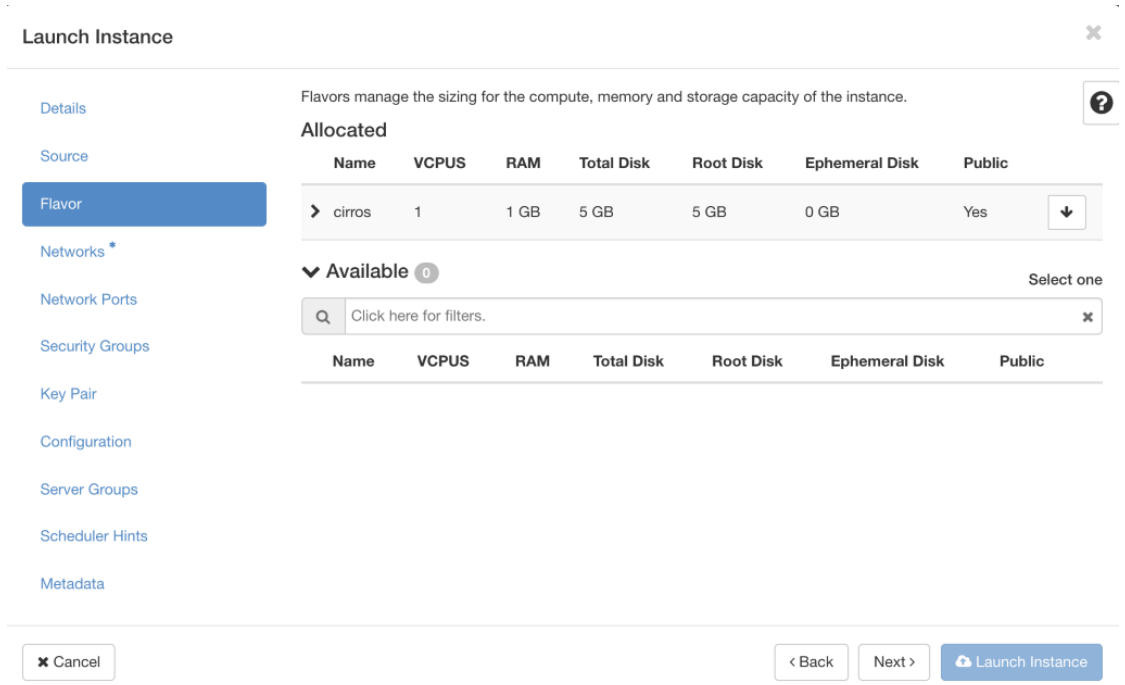


Demo3 – OpenStack VM, BMS間接続

Uploadしたcirros imageを選択

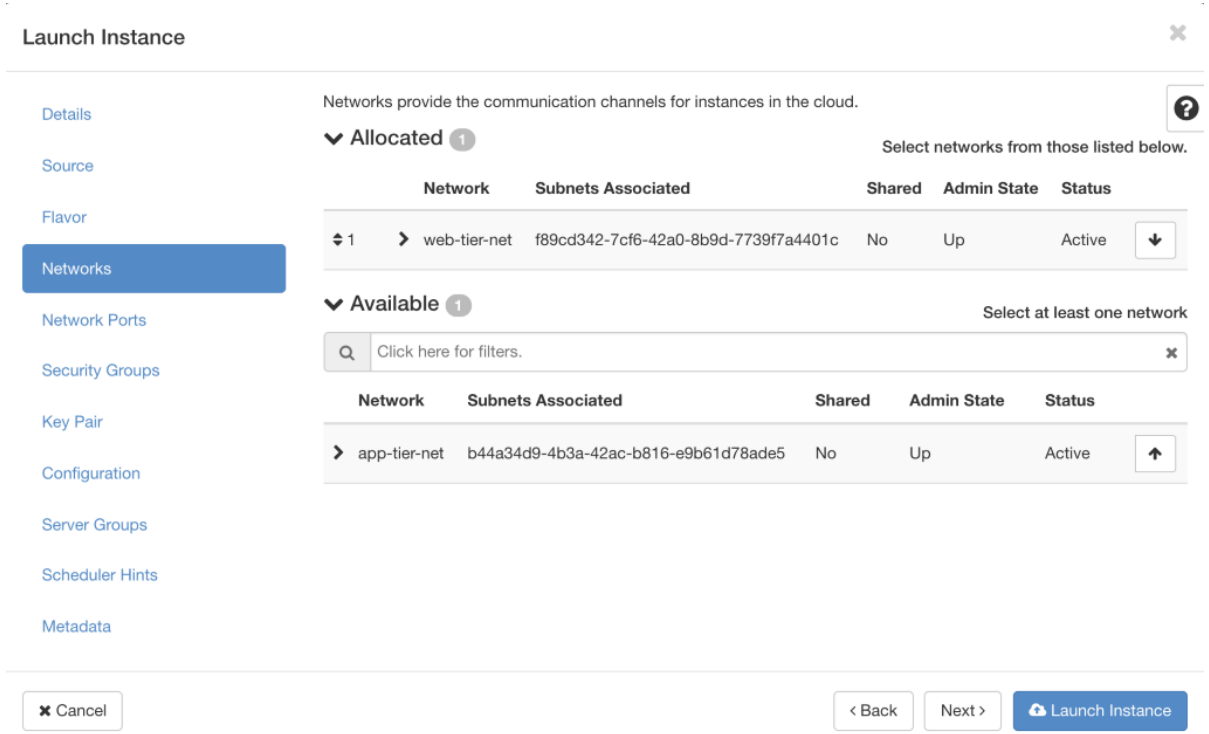


作成したcirros flavorを選択



Demo3 – OpenStack VM, BMS間接続

InstanceのSubnetとして、web-tier-netを指定し、Launch Instance




Thank you
JUNIPER NETWORKS
Engineering
Simplicity