ATTENTION: THIS IS A POC/LAB Solution Work WHICH MAY NOT BE a SLA BACKED SOLUTION BY RED HAT .<Author F.E. NAR>

How to Stretch an OCP cluster from AWS to On-premise.

1.0 Introduction

We will be adding additional remote worker node @on-premise to an existing ocp@aws (release 4.7.19 ipi) cluster. The idea is to have a local worker node close to backend at premises as well as service consumers with low latency access.

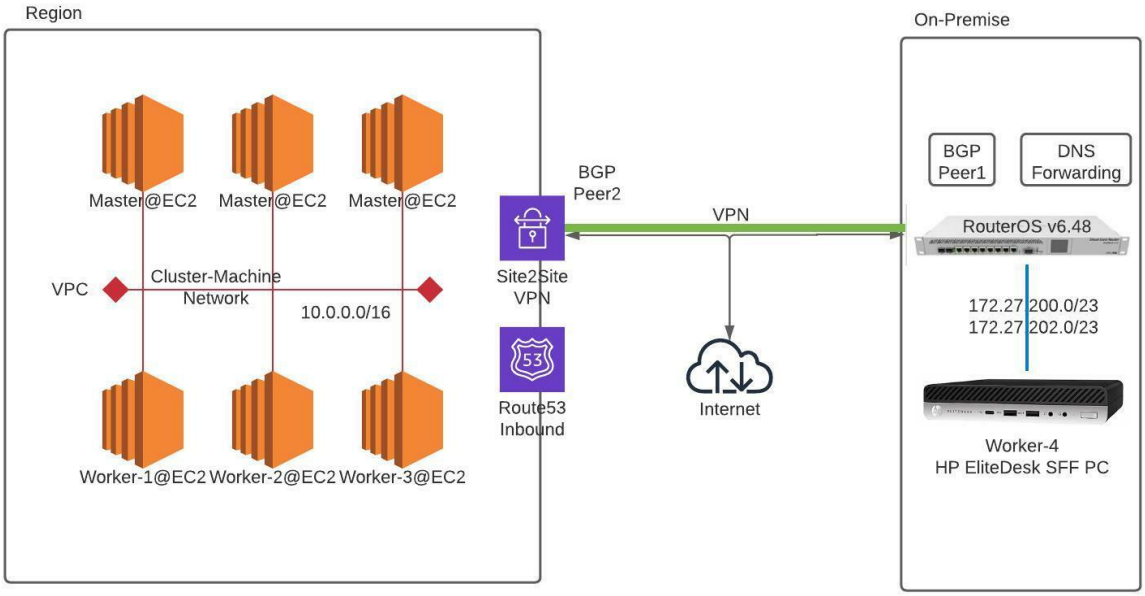
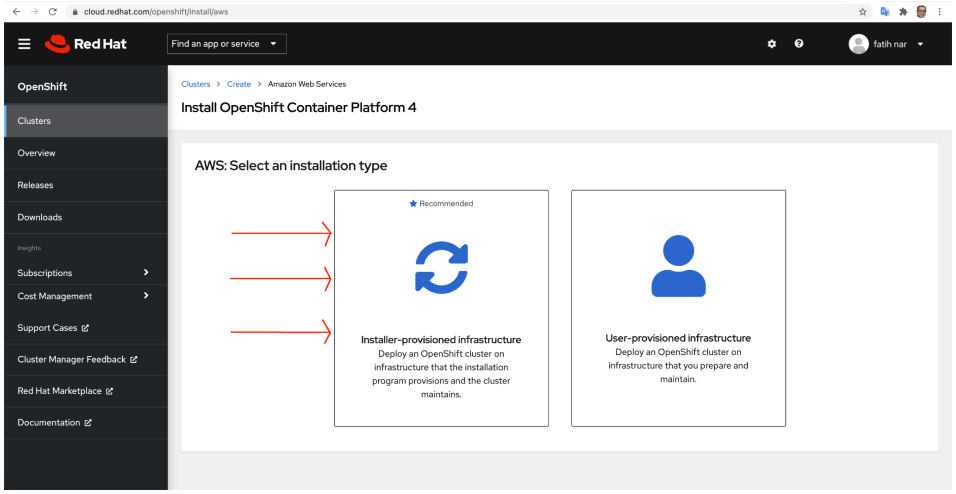


Figure-1 Solution TestBed Diagram

2.0 Prerequisites:

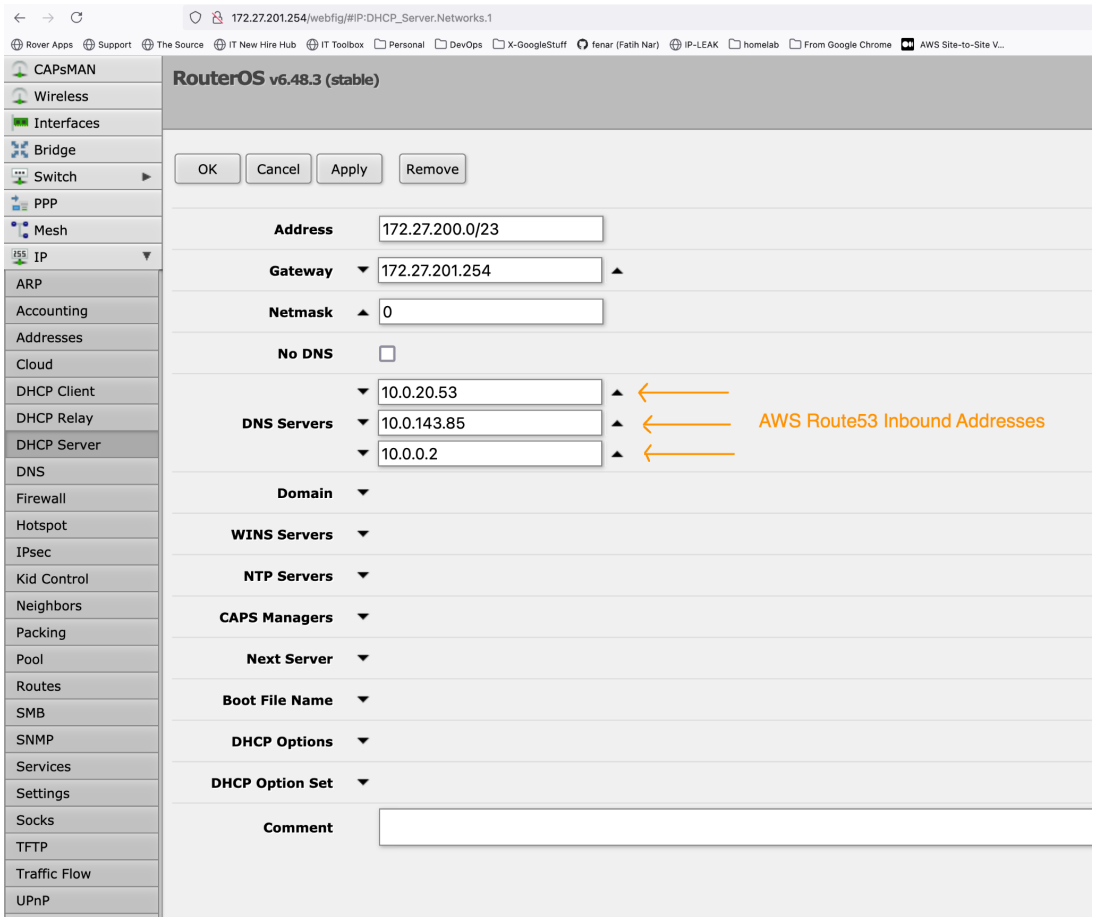
I. OCP IPI deployed on AWS. Reference: Link1, Link2.



ATTENTION: THIS IS A POC/LAB Solution Work WHICH MAY NOT BE a SLA BACKED SOLUTION BY RED HAT .<Author F.E. NAR>

II. On-premise site route configurable (example vpn guidance reference: Link2) with;

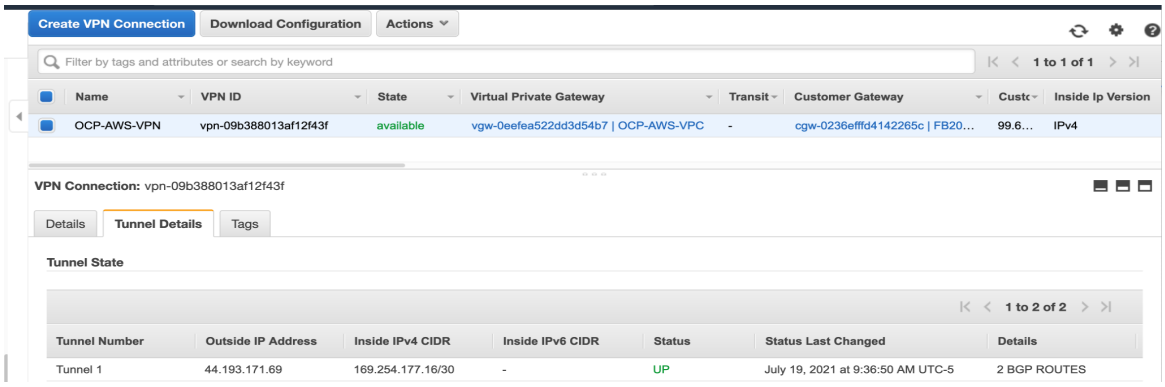
- ❖ Dhcp server with host dns provisioning pointing aws route53 inbound

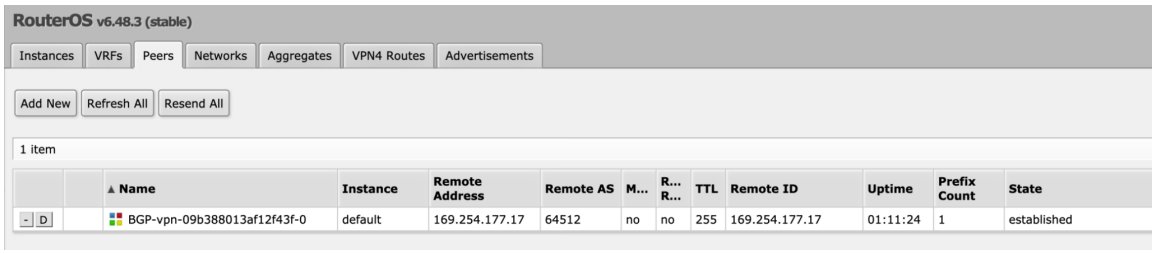


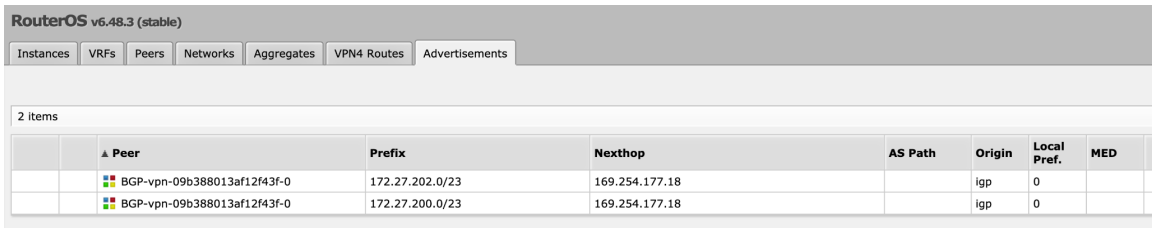
ATTENTION: THIS IS A POC/LAB Solution Work WHICH MAY NOT BE a SLA BACKED SOLUTION BY RED HAT .<Author F.E. NAR>

3.0 Network Readiness on AWS and On-Premise:

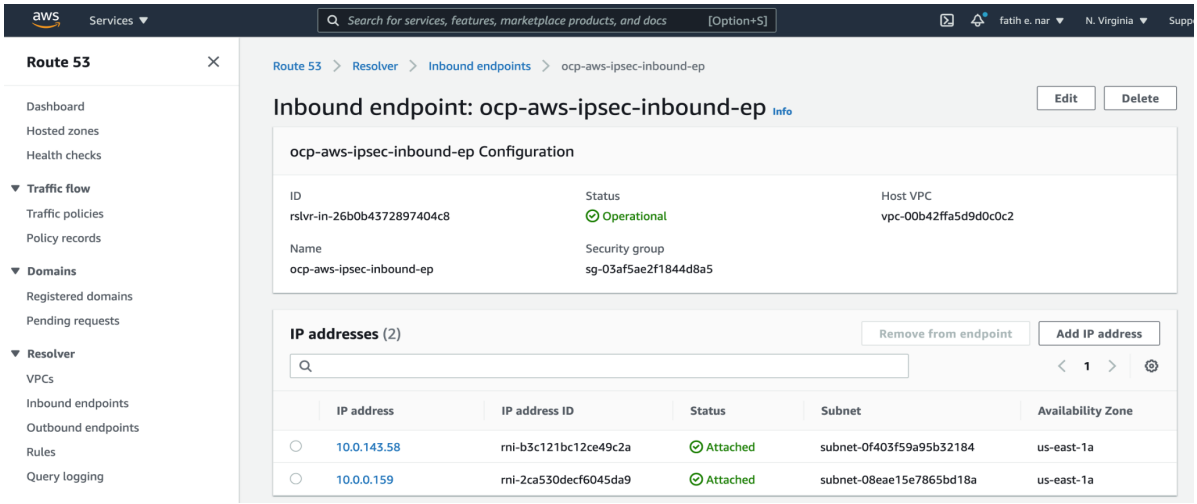
A. Create and attach VPN to a VPC where OCP Cluster deployed. Verify BP Peer connection established with on-premise router and route advertisement exchanged.





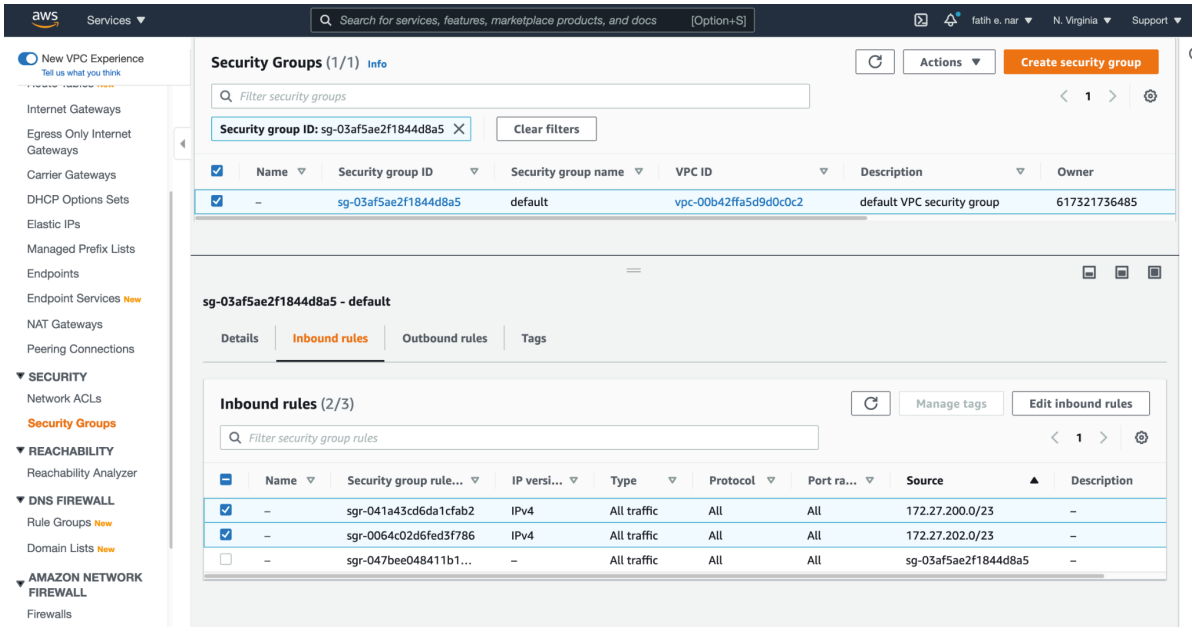


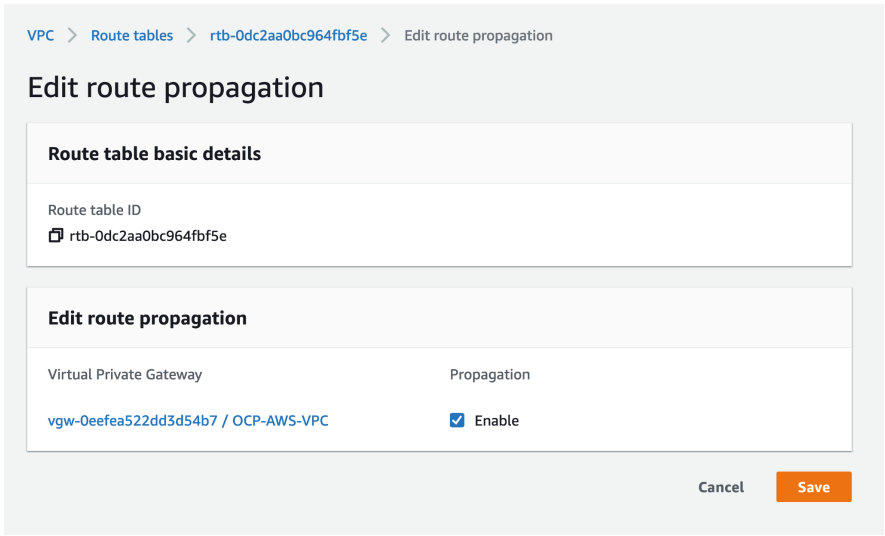
B. Configure Route 53 Inbound for remote worker node (rwn) to access route 53 private zone records.



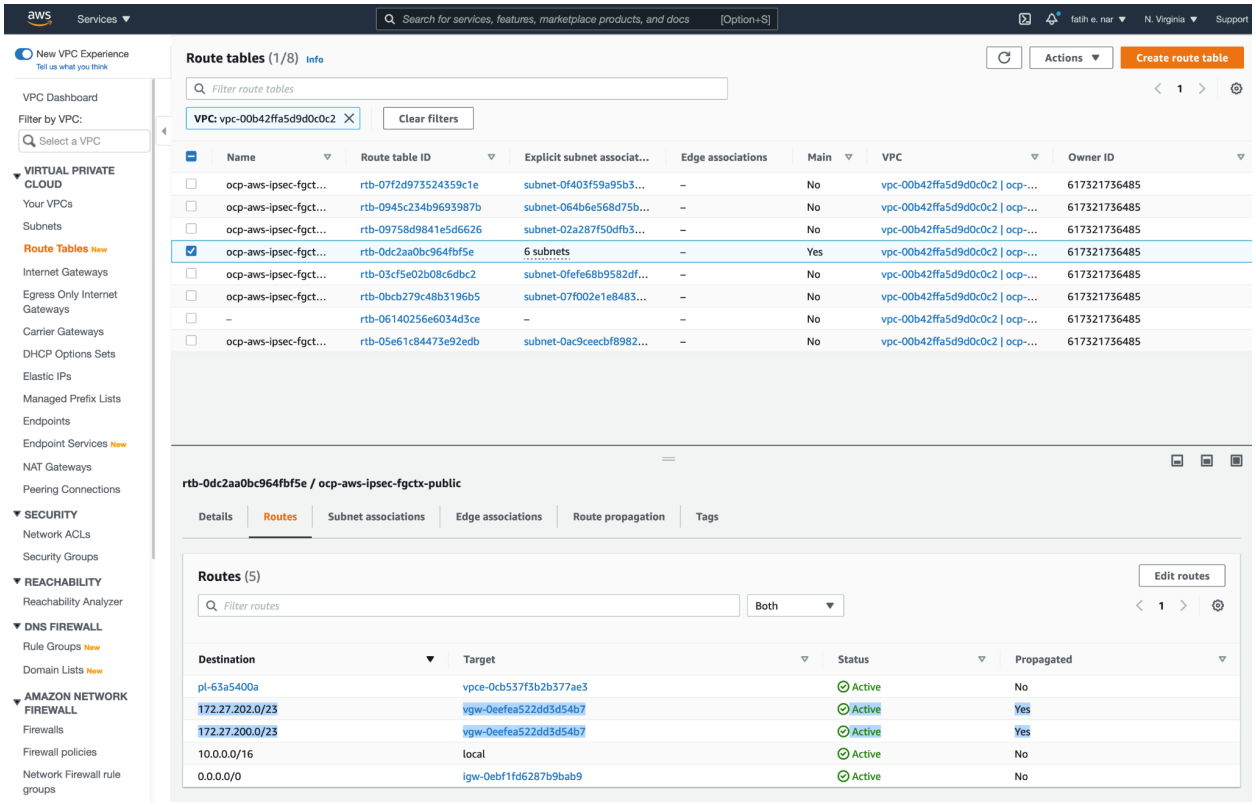
ATTENTION: THIS IS A POC/LAB Solution Work WHICH MAY NOT BE a SLA BACKED SOLUTION BY RED HAT .<Author F.E. NAR>

C. Configure aws vpc subnets, security groups for return path to on-premise routers.





ATTENTION: THIS IS A POC/LAB Solution Work WHICH MAY NOT BE a SLA BACKED SOLUTION BY RED HAT .<Author F.E. NAR>



4.0 Build your Remote Worker Node (RWN) on Premise

a. Boot your remote worker node with RHCOS Live ISO Image. (Ref: Repo)

```
[core@ip-172-27-201-49 ~]$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s25: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether f4:4d:30:65:a0:64 brd ff:ff:ff:ff:ff:ff
    inet 172.27.201.49/23 brd 172.27.201.255 scope global dynamic noprefixroute enp0s25
       valid_lft 376sec preferred_lft 376sec
    inet6 fe80::f823:f9ea:c63c:c427/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
[core@ip-172-27-201-49 ~]$ netstat -arn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         172.27.201.254  0.0.0.0         UG        0 0          0 enp0s25
172.27.200.0    0.0.0.0         255.255.254.0   U         0 0          0 enp0s25
[core@ip-172-27-201-49 ~]$ cat /etc/resolv.conf
# Generated by NetworkManager
search ec2.internal
nameserver 10.0.143.58
nameserver 10.0.0.159
nameserver 10.0.0.2
[core@ip-172-27-201-49 ~]$ hostname -f
ip-172-27-201-49.ec2.internal
```

On-Premise Site Router Address

AWS Route53 Inbound IP

AWS EC2 Reverse DNS Provided Hostname for RWN

b. Retrieve your worker ignition file

“curl -H "Accept: application/vnd.coreos.ignition+json; version=3.2.0" -k -o worker.ign https://api-int.<clustername>.<domain-name>:22623/config/worker”

```
[core@ip-172-27-201-49 ~]$ curl -H "Accept: application/vnd.coreos.ignition+json; version=3.2.0" -k -o worker.ign https://api-int.ocp-aws-ipsec.narlabs.io:22623/config/worker
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  184k  100  184k    0     0   389k      0 --:--:-- --:--:-- --:--:--  389k
[core@ip-172-27-201-49 ~]$ ll
total 188
-rw-rw-r--. 1 core core 189103 Jul 19 15:17 worker.ign
```

ATTENTION: THIS IS A POC/LAB Solution Work WHICH MAY NOT BE a SLA BACKED SOLUTION BY RED HAT .<Author F.E. NAR>

c. Bootstrap (install -> reboot) your worker with downloaded ignition file.

```
[core@ip-172-27-201-49 ~]$ ll
total 188
-rw-rw-r--. 1 core core 189103 Jul 19 15:17 worker.ign
[core@ip-172-27-201-49 ~]$ sudo coreos-installer install /dev/sda --ignition-file worker.ign
Installing Red Hat Enterprise Linux CoreOS 47.83.202105220305-0 (Ootpa) x86_64 (512-byte sectors)
> Read disk 3.5 GiB/3.5 GiB (100%)
Writing Ignition config
Install complete.
[core@ip-172-27-201-49 ~]$ sudo reboot now
```

5.0 Admit your new worker node to cluster

A. Verify & correct your Kubelet.service and 00-multus.conf compare -> guide.

```
Description=Kubernetes Kubelet
Wants=rpc-statd.service network-online.target crio.service
After=network-online.target crio.service
After=ostree-finalize-staged.service

[Service]
Type=notify
ExecStartPre=/bin/mkdir --parents /etc/kubernetes/manifests
ExecStartPre=/bin/rm -f /var/lib/kubelet/cpu_manager_state
EnvironmentFile=/etc/os-release
EnvironmentFile=-/etc/kubernetes/kubelet-workaround
EnvironmentFile=-/etc/kubernetes/kubelet-env

ExecStart=/usr/bin/hyperkube \
    kubelet \
      --config=/etc/kubernetes/kubelet.conf \
      --bootstrap-kubeconfig=/etc/kubernetes/kubeconfig \
      --kubeconfig=/var/lib/kubelet/kubeconfig \
      --container-runtime=remote \
      --container-runtime-endpoint=/var/run/crio/crio.sock \
      --runtime-cgroups=/system.slice/crio.service \
      --node-labels=node-role.kubernetes.io/worker,node.openshift.io/os_id=${ID} \
      --node-ip=${KUBELET_NODE_IP} \
      --minimum-container-ttl-duration=6m0s \
      --volume-plugin-dir=/etc/kubernetes/kubelet-plugins/volume/exec \
      --pod-infra-container-image=quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:4a2f77cfc83e981b62edb84b58c6520a09b6ea11e91e635ae91b5d56cf33873a \
      --v=${KUBELET_LOG_LEVEL}

Restart=always
RestartSec=10

[Install]
WantedBy=multi-user.target
[core@ip-172-27-201-49 ~]$ cat /etc/kubernetes/cni/net.d/00-multus.conf
{ "cniVersion": "0.3.1", "name": "multus-cni-network", "type": "multus", "namespaceIsolation": true, "logLevel": "verbose", "binDir": "/opt/multus/bin", "readine
ssindicatorfile": "/var/run/multus/cni/net.d/10-ovn-kubernetes.conf", "kubeconfig": "/etc/kubernetes/cni/net.d/multus.d/multus.kubeconfig", "delegates": [ {"cniV
ersion":"0.4.0","name":"ovn-kubernetes","type":"ovn-k8s-cni-overlay","ipam":{},"dns":{},"logFile":"/var/log/ovn-kubernetes/ovn-k8s-cni-overlay.log","logLevel":"4
","logfile-maxsize":100,"logfile-maxbackups":5,"logfile-maxage":5} ] }
[core@ip-172-27-201-49 ~]$
```

B. Approve Pending CSRs.

```
Every 2.0s: oc get csr                                                                                    jumpserver.narlabs.io: Mon Jul 19 10:37:42 2021

NAME        AGE     SIGNERNAME                                    REQUESTOR                                                                   CONDITION
csr-78hnv   2m15s   kubernetes.io/kubelet-serving                 system:node:ip-172-27-201-49.ec2.internal                                    Approved,Issued
csr-c49fg   4m20s   kubernetes.io/kube-apiserver-client-kubelet   system:serviceaccount:openshift-machine-config-operator:node-bootstrapper   Approved,Issued
```

```
fenar@jumpserver:~ (ssh)
[fenar@jumpserver ~]$ oc get csr -o name | xargs oc adm certificate approve
certificatesigningrequest.certificates.k8s.io/csr-78hnv approved
certificatesigningrequest.certificates.k8s.io/csr-c49fg approved
```

C. Enlist new worker node

```
[fenar@jumpserver ~]$ oc get nodes
NAME                            STATUS   ROLES    AGE     VERSION
ip-10-0-131-121.ec2.internal    Ready    worker   156m    v1.20.0+87cc9a4
ip-10-0-136-66.ec2.internal     Ready    master   165m    v1.20.0+87cc9a4
ip-10-0-145-15.ec2.internal     Ready    worker   155m    v1.20.0+87cc9a4
ip-10-0-151-137.ec2.internal    Ready    master   165m    v1.20.0+87cc9a4
ip-10-0-160-201.ec2.internal    Ready    master   165m    v1.20.0+87cc9a4
ip-10-0-167-91.ec2.internal     Ready    worker   156m    v1.20.0+87cc9a4
ip-172-27-201-49.ec2.internal   Ready    worker   4m22s   v1.20.0+87cc9a4
[fenar@jumpserver ~]$ oc debug node/ip-172-27-201-49.ec2.internal
Starting pod/ip-172-27-201-49ec2internal-debug ...
To use host binaries, run `chroot /host`
Pod IP: 172.27.201.49
If you don't see a command prompt, try pressing enter.
sh-4.4# chroot /host
sh-4.4# netstat -arn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         172.27.201.254  0.0.0.0         UG        0 0          0 br-ex
10.128.0.0      10.128.4.1      255.252.0.0     UG        0 0          0 ovn-k8s-mp0
10.128.4.0      0.0.0.0         255.255.254.0   U         0 0          0 ovn-k8s-mp0
169.254.0.0     0.0.0.0         255.255.240.0   U         0 0          0 ovn-k8s-gw0
172.27.200.0    0.0.0.0         255.255.254.0   U         0 0          0 br-ex
172.30.0.0      10.128.4.1      255.255.0.0     UG        0 0          0 ovn-k8s-mp0
```

On-Premise Site Router

d. [Optional] See whats running on new rwn:

```
[core@ip-172-27-201-49 ~]$ sudo crictl ps
CONTAINER      IMAGE                                                                                                                     CREATED        STATE    NAME                            ATTEMPT  POD ID
de988e015f5ae  679649b39d5e4b3ac39569b37ac803e8febaf418a8c82fd60b4d4affc80fde4e                                                          5 minutes ago  Running  container-00                    0        44557257a593d
5a2bdb5edeff6  1c005219b608388d6fab14276d295a66adf44e1c71ff8623a1874d789681bffa                                                          9 minutes ago  Running  ovn-ipsec                       0        1889a7159cee8
919ec234ed562  quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:661c4a573b867366ec86417a4411cff0bc4f4471ea0f2212e42f05f509855c02   9 minutes ago  Running  dns-node-resolver               0        2211689d0c2d4
a35e85c7afe55  ce9c13ed8f0b91dbb86bd971d0044fb537f8cc90eee1a3fc2b2df739618f7f2b                                                          9 minutes ago  Running  kube-rbac-proxy                 0        2211689d0c2d4
aa8b30f09e764  ce9c13ed8f0b91dbb86bd971d0044fb537f8cc90eee1a3fc2b2df739618f7f2b                                                          9 minutes ago  Running  kube-rbac-proxy                 0        326ad7b1fd8aa
c5e3ac35f5e7b  quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:6fb461f48f59f44d6275fce9a434fb317a796851a7a0798eebc4e2dd2ee9316f   9 minutes ago  Running  dns                             0        2211689d0c2d4
78714bc86acb3  quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:a8fd25e1e0998d03e83e20505b855f41d41b3ffccbfe08b0f355ddc2052f4a87   9 minutes ago  Running  network-metrics-daemon          0        326ad7b1fd8aa
808c3d8f44428  quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:a9db83f67aa4389811bad29af878d038e18bc39f63673fe77fe30f9bf1bd97de   9 minutes ago  Running  hello-openshift-canary          0        824ac4b18a622
615713edd6a35  quay.io/openshift-release-dev/ocp-v4.0-art-dev@sha256:a9db83f67aa4389811bad29af878d038e18bc39f63673fe77fe30f9bf1bd97de   9 minutes ago  Running  network-check-target-container  0        2430355b0f228
3b3ab0b972eb7  b4b60824ab49e23b1d77776b6165b18692121a311cb867c4a21ba4a6578ba95d                                                          9 minutes ago  Running  kube-multus                     0        fb72e0acc7ba4
639cc378f04f8  ce9c13ed8f0b91dbb86bd971d0044fb537f8cc90eee1a3fc2b2df739618f7f2b                                                          9 minutes ago  Running  kube-rbac-proxy                 0        7910877269b52
75053ac64d7eb  a36d93743899391cb187ccbeb8c3ec9916f223abac65d4b786a4759bb6fceb84                                                          9 minutes ago  Running  node-exporter                   0        7910877269b52
9792793f521cd  da4e81f44b3c4e91d52f0e88c0a0358844922e1882a5257e9c9682a4ffe22329                                                          9 minutes ago  Running  csi-liveness-probe              0        30a3528751687
a0d1e6a86822d  202b33e23042b8d7691cd34d4bf6c97dd14ea8be37f7b2c5044909d3d5cd4cf2                                                          9 minutes ago  Running  node-ca                         0        3e797044005ca
2f1d8a2afc1cc  485ea7313c8214d1e8255b9aa2314e1455b7f2087053977a6fe79129a44e2882                                                          9 minutes ago  Running  csi-node-driver-registrar       0        30a3528751687
048ac7550b3de  d490e32bd94696e5260be24c17cc3bb3d360224c90bccc07b32effc3f649ebd3                                                          9 minutes ago  Running  oauth-proxy                     0        de554b2501501
1854c04dcc629  1c005219b608388d6fab14276d295a66adf44e1c71ff8623a1874d789681bffa                                                          9 minutes ago  Running  ovnkube-node                    0        b25c0875c68f3
7079a9ae4d14a  2f6049ff50722f9e10faf245ff201532255e244c397cfa30791ddabb1bc45af2                                                          9 minutes ago  Running  tuned                           0        b21cbabfbd42c
6a33a8a204c79  ce9c13ed8f0b91dbb86bd971d0044fb537f8cc90eee1a3fc2b2df739618f7f2b                                                          9 minutes ago  Running  kube-rbac-proxy                 0        b25c0875c68f3
d9ee93402a80e  810143c8f96eaf5bebccc95e0e9f19987b4f48b6169156e84d7cb23a82a7498e                                                          9 minutes ago  Running  machine-config-daemon           0        de554b2501501
e5a36a3dccfb6  1c005219b608388d6fab14276d295a66adf44e1c71ff8623a1874d789681bffa                                                          9 minutes ago  Running  ovn-controller                  0        b25c0875c68f3
```

Side Note: If you plan to deploy stateful workloads (i.e. use of persistent volumes) , you need to create a new storage class that points to the local disk (you can leverage local storage operator from the operator hub).