OpenShift 4.17 Bare Metal Static IP Installation Guide

Prerequisites

Ensure these are ready before starting:

- Red Hat account with valid subscription
- Pull secret from https://cloud.redhat.com/openshift/install
- SSH key pair for cluster access
- Static IP addresses for all nodes
- Internet connectivity for initial downloads

Network Configuration

Plan and document these IP assignments before starting - you'll need them throughout the installation.

```
Network: 192.168.7.0/24
Gateway: 192.168.7.1
DNS: 192.168.7.77

Nodes:
  Helper/Registry: 192.168.7.77
  Bootstrap: 192.168.7.20
  Master-0: 192.168.7.21
  Master-1: 192.168.7.22
  Master-2: 192.168.7.23
  Worker-0: 192.168.7.11
  Worker-1: 192.168.7.12
  API VIP: 192.168.7.100
  Apps VIP: 192.168.7.101
```

Step 1: Setup Helper Node

```
# Install CentOS/RHEL 8 with static IP 192.168.7.77
ssh root@192.168.7.77

# Update and configure system
yum update -y
hostnamectl set-hostname registry.eiab.us.eclub.com

# Configure firewall
systemctl enable firewalld --now
firewall-cmd --permanent --add-port={53/tcp,53/udp,67/udp,69/udp,80/tcp,443/tcp,5000/tcp,6443/tcp,8080/tcp,9000/tcp,22623/tcp}
firewall-cmd --reload

# Install required packages
yum -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-$(rpm -E %rhel).noarch.rpm
yum -y install ansible git wget curl podman httpd-tools jq bind-utils net-tools skopeo

# Clone helper node repository
git clone https://github.com/redhat-cop/ocp4-helpernode
cd ocp4-helpernode
```

Step 2: Download OpenShift Components

```
# Set version
export OCP_VERSION=4.17.0
export RHCOS_VERSION=417.94.202411051319-0

# Download OpenShift client and installer
wget https://mirror.openshift.com/pub/openshift-v4/clients/ocp/${OCP_VERSION}/openshift-client-linux-${OCP_VERSION}.tar.gz
tar xvf openshift-client-linux-${OCP_VERSION}.tar.gz -C /usr/local/bin/
wget https://mirror.openshift.com/pub/openshift-v4/clients/ocp/${OCP_VERSION}/openshift-install-linux-${OCP_VERSION}.tar.gz
tar xvf openshift-install-linux-${OCP_VERSION}.tar.gz -C /usr/local/bin/
rm -f *.tar.gz

# Download RHCOS images
mkdir -p /var/www/html/install
wget -O /var/www/html/install/vmlinuz https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/4.17/latest/rhcos-${RHCOS_VER
wget -O /var/www/html/install/initramfs.img https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/4.17/latest/rhcos-${RHC
wget -O /var/www/html/install/rootfs.img https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/4.17/latest/rhcos-${RHCOS_
wget -O ~/rhcos-live.iso https://mirror.openshift.com/pub/openshift-v4/dependencies/rhcos/4.17/latest/rhcos-${RHCOS_VERSION}-live.x8
```

Step 3: Setup Local Registry

```
# Create registry directories and certificates
mkdir -p /opt/registry/{auth,certs,data}

openssl req -newkey rsa:4096 -nodes -sha256 \
  -keyout /opt/registry/certs/domain.key \
  -x509 -days 365 \
  -out /opt/registry/certs/domain.crt \
  -subj "/CN=registry.eiab.us.eclub.com"

cp /opt/registry/certs/domain.crt /etc/pki/ca-trust/source/anchors/
update-ca-trust

# Setup authentication
htpasswd -bBc /opt/registry/auth/htpasswd admin redhat123

# Start registry
podman run -d --name mirror-registry -p 5000:5000 --restart=always \
  -v /opt/registry/data:/var/lib/registry:z \
  -v /opt/registry/auth:/auth:z \
  -v /opt/registry/certs:/certs:z \
  -e REGISTRY_AUTH=htpasswd \
  -e REGISTRY_AUTH_HTPASSWD_REALM="Registry Realm" \
  -e REGISTRY_AUTH_HTPASSWD_PATH=/auth/htpasswd \
  -e REGISTRY_HTTP_TLS_CERTIFICATE=/certs/domain.crt \
  -e REGISTRY_HTTP_TLS_KEY=/certs/domain.key \
  docker.io/library/registry:2
```

Step 4: Configure Helper Node Services

```
# Create vars-static.yaml
cat > vars-static.yaml <<EOF
---
disk: vda
helper:
  name: "helper"
  ipaddr: "192.168.7.77"
dns:
  domain: "us.eclub.com"
  clusterid: "eiab"
  forwarder1: "8.8.8.8"
  forwarder2: "8.8.4.4"
bootstrap:
  name: "bootstrap"
  ipaddr: "192.168.7.20"
masters:
  - name: "master0"
    ipaddr: "192.168.7.21"
  - name: "master1"
    ipaddr: "192.168.7.22"
  - name: "master2"
    ipaddr: "192.168.7.23"
workers:
  - name: "worker0"
    ipaddr: "192.168.7.11"
  - name: "worker1"
    ipaddr: "192.168.7.12"
EOF

# Run helper node setup
ansible-playbook -e @vars-static.yaml -e staticips=true tasks/main.yml

# Verify services
/usr/local/bin/helpernodecheck
```

Step 5: Mirror OpenShift Images

```
 # Download pull secret from https://cloud.redhat.com/openshift/install
mkdir -p ~/.openshift
# Save pull secret as ~/.openshift/pull-secret

# Add local registry to pull secret
podman login --authfile ~/.openshift/pull-secret registry.eiab.us.eclub.com:5000 -u admin -p redhat123

# Download oc-mirror
wget https://mirror.openshift.com/pub/openshift-v4/clients/ocp/${OCP_VERSION}/oc-mirror.tar.gz
tar xvf oc-mirror.tar.gz -C /usr/local/bin/
chmod +x /usr/local/bin/oc-mirror
rm -f oc-mirror.tar.gz

# Create image set config
cat > imageset-config.yaml <<EOF
kind: ImageSetConfiguration
apiVersion: mirror.openshift.io/v1alpha2
storageConfig:
  registry:
    imageURL: registry.eiab.us.eclub.com:5000/eiab/openshift4
    skipTLS: false
mirror:
  platform:
    channels:
    - name: stable-4.17
      type: ocp
      minVersion: 4.17.0
      maxVersion: 4.17.0
EOF

# Mirror images
oc-mirror --config=imageset-config.yaml docker://registry.eiab.us.eclub.com:5000
```

Step 6: Generate Ignition Files

```
# Create working directory
mkdir ~/ocp-install
cd ~/ocp-install

# Create install-config.yaml
cat > install-config.yaml <<EOF
apiVersion: v1
baseDomain: us.eclub.com
compute:
- hyperthreading: Enabled
  name: worker
  replicas: 2
controlPlane:
  hyperthreading: Enabled
  name: master
  replicas: 3
metadata:
  name: eiab
networking:
  clusterNetwork:
  - cidr: 10.128.0.0/14
    hostPrefix: 23
  networkType: OVNKubernetes
  serviceNetwork:
  - 172.30.0.0/16
platform:
  none: {}
pullSecret: '$(< ~/.openshift/pull-secret)'
sshKey: '$(< ~/.ssh/helper_rsa.pub)'
imageContentSources:
- mirrors:
  - registry.eiab.us.eclub.com:5000/eiab/openshift4
  source: quay.io/openshift-release-dev/ocp-release
- mirrors:
  - registry.eiab.us.eclub.com:5000/eiab/openshift4
  source: quay.io/openshift-release-dev/ocp-v4.0-art-dev
additionalTrustBundle: |
$(sed 's/^/  /' < /opt/registry/certs/domain.crt)
EOF

# Generate manifests and ignition configs
cp install-config.yaml install-config.yaml.backup
openshift-install create manifests
sed -i 's/mastersSchedulable: true/mastersSchedulable: false/g' manifests/cluster-scheduler-02-config.yml
openshift-install create ignition-configs

# Copy to web server
cp ~/ocp-install/*.ign /var/www/html/ignition/
chmod 644 /var/www/html/ignition/*.ign
restorecon -vR /var/www/html/
```

Step 7: Install RHCOS on Nodes

Boot each node from RHCOS ISO and press TAB at boot menu. Add kernel parameters:

Bootstrap Node

```
ip=192.168.7.20::192.168.7.1:255.255.255.0:bootstrap.eiab.us.eclub.com:ens192:none nameserver=192.168.7.77 coreos.inst.install_dev=\
```

Master-0

```
ip=192.168.7.21::192.168.7.1:255.255.255.0:master0.eiab.us.eclub.com:ens192:none nameserver=192.168.7.77 coreos.inst.install_dev=vda
```

Master-1

```
ip=192.168.7.22::192.168.7.1:255.255.255.0:master1.eiab.us.eclub.com:ens192:none nameserver=192.168.7.77 coreos.inst.install_dev=vda
```

Master-2

```
ip=192.168.7.23::192.168.7.1:255.255.255.0:master2.eiab.us.eclub.com:ens192:none nameserver=192.168.7.77 coreos.inst.install_dev=vda
```

Worker-0

```
ip=192.168.7.11::192.168.7.1:255.255.255.0:worker0.eiab.us.eclub.com:ens192:none nameserver=192.168.7.77 coreos.inst.install_dev=vda
```

Worker-1

```
ip=192.168.7.12::192.168.7.1:255.255.255.0:worker1.eiab.us.eclub.com:ens192:none nameserver=192.168.7.77 coreos.inst.install_dev=vda
```

Step 8: Bootstrap and Install Cluster

```
# Monitor bootstrap progress (in separate terminal windows)
# Terminal 1 - Bootstrap monitoring
openshift-install --dir ~/ocp-install/ wait-for bootstrap-complete --log-level=debug

# Terminal 2 - Check status dashboard (optional)
firefox http://192.168.7.77:9000

# Terminal 3 - SSH to bootstrap to monitor logs (optional)
ssh core@bootstrap.eiab.us.eclub.com
sudo journalctl -b -f -u release-image.service -u bootkube.service

# Once bootstrap complete, remove bootstrap node
# Power off bootstrap VM

# Setup kubeconfig
export KUBECONFIG=~/ocp-install/auth/kubeconfig

# Verify API access
oc whoami
oc get nodes

# Approve worker CSRs
oc get csr -o go-template='{{range .items}}{{if not .status}}{{.metadata.name}}{{"\n"}}{{end}}{{end}}' | xargs oc adm certificate ap

# Watch for additional CSRs (run multiple times)
watch oc get csr

# Configure image registry
oc patch configs.imageregistry.operator.openshift.io cluster --type merge --patch '{"spec":{"managementState":"Managed"}}'
oc patch configs.imageregistry.operator.openshift.io cluster --type merge --patch '{"spec":{"storage":{"emptyDir":{}}}}'

# Monitor operator rollout
watch oc get clusteroperators

# Wait for installation to complete
openshift-install --dir ~/ocp-install/ wait-for install-complete
```

Step 9: Access Cluster

```
# Get console URL and credentials
oc whoami --show-console
cat ~/ocp-install/auth/kubeadmin-password

# Verify cluster
oc get nodes
oc get co
oc get clusterversion

# Scale ingress controller
oc patch --namespace=openshift-ingress-operator --patch='{"spec": {"replicas": 3}}' --type=merge ingresscontroller/default
```

Quick Verification Commands

```
# Check cluster health
oc get nodes
oc get co | grep -v "True.*False.*False"
oc get pods --all-namespaces | grep -v Running | grep -v Completed

# Upgrade cluster (if needed)
oc adm upgrade --to-latest
```

Additional Configurations

```
# Configure NTP synchronization (optional)
cat > ntp-machineconfig.yaml <<EOF
apiVersion: machineconfiguration.openshift.io/v1
kind: MachineConfig
metadata:
  labels:
    machineconfiguration.openshift.io/role: worker
  name: 50-worker-chrony
spec:
  config:
    ignition:
      version: 3.2.0
    storage:
      files:
      - contents:
          source: data:text/plain;charset=utf-8;base64,$(echo 'server 192.168.7.77 iburst
driftfile /var/lib/chrony/drift
makestep 1.0 3
rtcsync
logdir /var/log/chrony' | base64 -w 0)
        filesystem: root
        mode: 0644
        path: /etc/chrony.conf
EOF
oc apply -f ntp-machineconfig.yaml

# Configure cluster autoscaling (optional)
oc patch clusterautoscaler default --type merge --patch '{"spec":{"scaleDown":{"enabled":true,"delayAfterAdd":"10m","delayAfterDelet

# Set default node selector for user workloads
oc patch scheduler cluster --type merge --patch '{"spec":{"defaultNodeSelector":"node-role.kubernetes.io/worker="}}'
```

Troubleshooting

```
# Check specific operator
oc logs -n openshift-cluster-version -l k8s-app=cluster-version-operator

# Debug node
oc debug node/<node-name>

# Check events
oc get events --all-namespaces --sort-by='.lastTimestamp' | tail -20

# Check certificate expiration
oc get csr -o json | jq -r '.items[] | select(.status.certificate == null) | .metadata.name'

# Force certificate renewal
oc adm certificate approve $(oc get csr -o json | jq -r '.items[] | select(.status.certificate == null) | .metadata.name')

# Check node logs
oc adm node-logs <node-name> --role=master
oc adm node-logs <node-name> --role=worker
```