ionir

Deployment Requirements

Version 2.4 for OCP

Contents

Ionir Software Requirements

RedHat Openshift Container Platform (OCP) Support

- Support for OCP is through an Operator which will be installed and controls the entire lifecycle of Ionir data platform.
- Ionir version 2.4 supports OCP 4.6.x.
- RHCOS as worker nodes OS (for RHEL support contact support@ionir.com)
- For more information visit RedHat Certified OperatorHub.

Ionir Operator

Ionir Certified Operator can be found on Openshift OperatorHub, please make sure its accessible for installation.
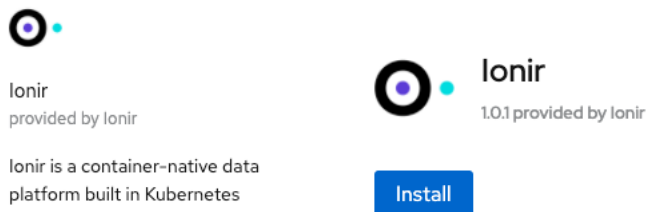
Ionir Machine Configuration

Download and run four (4) machine configuration files examples from Ionir-cloud GitHub –

https://github.com/ionir-cloud/deployments/tree/main/OpenShift/configuration/machine-configuration-files

Required Settings

SR-IOV network is an optional feature of an Openshift cluster. To make it work, it requires different components to be provisioned and configured accordingly. For best performance using SR-IOV we recommend that this setting is enabled in the system grub file **on each node**:

- For Intel processors - intel_iommu must be set to on (intel_iommu=on)
  - Example YAML here
- For AMD processors - amd_iommu must be set to on (amd_iommu=on)

The following additional modules must be loaded and running examples here

- NVME/TCP - nvme_tcp (modprobe nvme_tcp)
- VFIO - vfio-pci (modprobe vfio_pci)

The following parameters must be tuned as well:

- 2MB hugepages (vm.nr_hugepages) must be set to 1024 or higher
  - Example YAML here
- arp_ignore must be set to 1

Note: A reboot is mandatory as part of this process

Ionir Resource Requirements

Kubernetes Worker Nodes

The following are the **minimum resources required** to run Ionir Cloud native storage solution. Additional resources and capacity needed to run the customers apps and services on top per customer design.

CPU Type	Haswell or higher (Intel), EPYC or higher (AMD) (AVX2 instruction set required)
Number of physical Cores	8 Cores
Physical Memory	32GB
Network Interface	Ionir requires a dedicated virtual (using SR-IOV **recommended**) or physical NIC • Supported NIC manufacturers: ○ Mellanox - connectx-5 recommended ○ Intel ○ Broadcom • NIC must be 10Gbe or higher (25Gbe is recommended for production environments)
Media	• Up to 20 Local NVMe SSDs on each worker node • NVMe media device must be reported as an NVMe device by the Linux system (`lsblk/lspci`) • No filesystem should exist on the NVMe media • Each drive should be fully accessible by Ionir (all partitions will be deleted) • NVMe Media size - 512GB

Boot Media   The boot device must be an SSD with at least 128 GB of free space.

Physical Capacity Requirements

The minimum cluster size is determined by the minimum number of Ionir nodes that have media **installed locally.**

- For a 3-node deployment - 3.6 TB in total
- For a 5-node deployment and higher - 6TB in total

Networking Requirements

Ionir creates a high performance dedicated virtual network for storage traffic, and hence requires that the underlying network infrastructure provide sufficient performance:

- For production environments network speed must be 25 Gb or higher. 10 Gb may be used in non-production environments.
- Full IP connectivity is required between all nodes in the cluster.
- SR-IOV must be enabled on the NIC as best practice. Optionally use multiple physical adapters if applicable.
    - If SR-IOV is used, VFS must be enabled using the NIC management software to provide a second network interface
- Full IP connectivity is required between all nodes and **between clusters for mobility.**

Pre-Install Node Network Configuration

Ionir creates a dedicated high-speed data network between all worker nodes in the cluster. This network is used for communication between the Ionir pods and provides a high performance datapath. This network can be virtual using the same physical interface as that used by Kubernetes by enabling SR-IOV on the network adapters, or it can be configured using dedicated NICs.

> **Note:** Ionir datapath network must be configured with a unique subnet. This network does not require a default gateway and is internal only.

The following configuration is required for each worker node in the cluster:

- Set a second **static IP** for the node on the interface for the data path network (physical or virtual function). This datapath IP must be part of the datapath subnet.
- MTU must be set to 9000 or higher (Jumbo frames).
- A Kubernetes label must be set that states the datapath IP of the node must be added to **each worker node**. To set the label run the following command for each worker node:

```
kubectl label node <workerNodeName> datapath_ni=<NodeDatapathIP>
```

Other Requirements

Load Balancer

An external load balancer (such as MetalLB) is required on the Kubernetes cluster to provide an external accessible IP address for the Ionir platform in general and mobility between cluster specifically. This loadbalancer resource should be configurable at the Kubernetes level using standard services.

Example MetalLB deployment YAML can be found on ionir github repository here.

Note: When working with multiple clusters load balancers of different clusters must not overlap

Access to Install Images

Prior to installation contact sales support to get ***ionir license key.***

All nodes require direct access to an image registry (either private or public) to be able to pull the Ionir images and other 3rd party images.

NTP Requirements

It is highly recommended that the time on all nodes be synchronized using NTP.

Ionir Compatibility Check Tool

The compatibility check tool can run on any Linux machine that has kubectl [installed](#) and has a directory with the compatibility tool script and yaml files. A namespace called "ionir-compatibility-check" is created and will be deployed with a pod on each node (using daemonset) in the namespace.

To run the compatibility Check Tool, follow these steps:

1. Open the compatibility check directory in a Linux console.
2. Make sure kubectl is installed and that KUBECONFIG is configured to the kubernetes cluster
3. The compatibility check script expects to get the following parameters:

```
-i  : Install compatibility tool
-r  : [optional] image registry to pull the images from (default:
      quay.io/ionir)
-u  : username for the image registry
-p  : image registry password
-t  : image tag name
-s  : [installation type] Ionir's installation type, may be
      either 'minimal', for up to 4 nodes or 'scale', for a
      larger cluster
```

4. Run the ionir-compatibility-check.sh script to install the tool:

```
./ionir-compatibility-check.sh -i -r <registry> -u <user> -p
<password> -s [install type] -t <tag>
```

Here is an example on running the tool with user "ionir" and password "test" and size "minimal" using default quay.io/ionir public repo

```
./ionir-compatibility-check.sh -i -u ionir -p test -s minimal -t
v2.4
```

The test will verify that all nodes meet the minimal hardware and software requirements, that the cluster network meets Ionir requirements, and images can be pulled from the image registry.

Overall Result - The first section in the result is the overview status of the cluster. The following output indicates that the Kubernetes cluster is correctly configured for a successful Ionir installation.

Note: If any other result is shown, please contact technical support support@ionir.com