

Release Notes

Published
2023-07-13

Release Notes: Cloud-Native Contrail Networking 23.2

Table of Contents

Introduction | 1

What's New | 1

Tested Integrations | 4

Container Tags | 4

Open Issues | 5

Resolved Issues | 10

Requesting Technical Support | 10

Revision History | 12

Introduction

Juniper Cloud-Native Contrail® Networking (CN2) is a cloud-native SDN solution that provides advanced networking capabilities to containerized cloud networking environments. CN2 is optimized for Kubernetes-orchestrated environments and can be used to connect, isolate, and secure cloud workloads and services seamlessly across private, public, and hybrid clouds.

These release notes accompany Release 23.2 of CN2. They describe new features, limitations, platform compatibility requirements, known behavior, and resolved issues in CN2.

See the [Cloud-Native Contrail Networking \(CN2\)](#) page for a complete list of all CN2 documentation.

What's New

IN THIS SECTION

- [CN2 on Rancher RKE2 | 1](#)
- [CN2 on Upstream Kubernetes | 2](#)
- [Configure Kubernetes | 2](#)
- [Advanced Virtual Networking | 2](#)
- [Analytics | 3](#)
- [CN2 Pipelines | 4](#)

Learn about new features introduced in CN2 Release 23.2.

CN2 on Rancher RKE2

- Starting in Release 23.2, CN2 is supported on a Rancher RKE2 cluster. See the [Installation and Life Cycle Management Guide for Rancher RKE2](#).

CN2 on Upstream Kubernetes

- Starting in Release 23.2, CN2 is supported on Kubernetes v1.26.

Configure Kubernetes

- **Priority Classes**—Starting in Release 23.2, CN2 supports Priority Classes for critical CN2 components. CN2 introduces the `PriorityClass` object, which lets you map a priority, in the form of an integer value, to a priority class name. CN2's essential components use these default classes so that kube-scheduler prioritizes these pods for scheduling and resource allocation.

[See [Priority Classes for Critical Components](#)].

- **Multi-Cluster Pod Scheduling**—Starting in CN2 Release 23.2, CN2 supports network-aware pod scheduling for multi-cluster deployments. CN2 introduces the `MetricsConfig` controller and the `CentralCollector` controller. These controllers reconcile and manage a custom metrics collector CR and a central collector CR. These custom resources enable the `contrail-scheduler` to schedule multi-cluster pods based on important network metrics.

[See [Pod Scheduling for Multi-Cluster Deployments](#)].

Advanced Virtual Networking

- **Fast Convergence**—Starting in Release 23.2, CN2 supports Fast Convergence. CN2 provides an SDN solution that offers network virtualization at the compute node-level through overlay networking. In an SDN, failures can occur in the overlay or in the underlay. The vRouter detects, rectifies, and propagates any failure to the gateways by using health checks. Fast convergence improves the convergence time in case of failures in a cluster managed by CN2.

[See [Configure Fast Convergence in CN2](#)].

- **Graceful Restart and Long-Lived Graceful Restart**—Starting in Release 23.2, you can configure graceful restart and long-lived graceful restart (LLRG) in CN2. LLGR is a mechanism used to preserve routing details for a longer period of time in the event of a failed peer. Graceful restart and LLGR ensure that routes learnt are not immediately deleted and withdraw from advertised peers. Instead, the routes are kept and marked as stale. Consequently, if sessions come back up and routes are relearned, the overall impact to the network is minimized.

[See [Configure Graceful Restart and Long-Lived Graceful Restart](#)].

- **BFD Health Check for BGPaaS Sessions**—Starting in CN2 Release 23.2, you can configure Bidirectional Forwarding and Detection (BFD) health check for BGP as a Service (BGPaaS) sessions. When you configure BFD health check, you associate the health check service with a BGPaaS object. This association triggers the establishment of BFD sessions to all BGPaaS neighbors for that service. If the BFD session goes down, the resulting BGPaaS session terminates and the routes are withdrawn.

[See [Configure BFD Health Check for BGPaaS Sessions](#)].

- **Stickiness for Load-Balanced Flows**—Starting in Release 23.2, CN2 supports flow stickiness. Flow stickiness helps minimize flow remapping across ECMP groups in a load-balanced system. Flow stickiness reduces the flow being remapped and retains the flow with the original path when the ECMP group's member change. When a flow is affected by a member change, the vRouter reprograms the flow table and rebalances the flow.

[See [Stickiness for Load-Balanced Flows](#)].

Analytics

- **Extend TLS to Analytics**—Starting in Release 23.2, you can enable TLS certificates for analytics components in CN2. TLS is a security protocol used for certificate exchange, mutual authentication, and negotiating ciphers to secure the stream from potential tampering and eavesdropping. By default, the certificate and secrets for the control plane and vRouter are automatically generated in Contrail certificate manager. When you install the components with Helm, certificate manager automatically creates the certificates and secrets needed for each analytic component.
- **Flow-based traffic mirroring**—Starting in CN2 Release 23.2, CN2 can selectively mirror network traffic on the basis of flow when vRouter is in flow mode. This network traffic flow is specified by the security policy and is sent to the network analyzer that monitors and analyzes the data. The network analyzer is specified with `mirrorDestination` resource. It also supports the `mirrorDestination` resource present outside the cluster.

If the security policy defines `SecondaryAction` at the rule level, then flows matching the rules with `mirror destination` are mirrored.

[See [Flow-Based Mirroring](#)].

CN2 Pipelines

CN2 Pipelines is a CI/CD tool to enable GitOps-based workflows to automate CN2 configuration, testing, and qualification. CN2 Pipelines runs alongside CN2 clusters starting with CN2 Release 23.1 (Tech Preview). In Release 23.2, CN2 Pipelines supports customer container network functions (CNFs), auto-generates bearer token for authentication, discovers cluster nodes dynamically and uses discovered data during test execution.

[See [CN2 Pipelines for GitOps Guide](#)].

Tested Integrations

Starting in CN2 Release 23.1, Supported Platforms is now documented in [CN2 Tested Integrations](#). This document includes integrations fully tested and validated by Juniper, including tested NICs and other software components.

Container Tags

Container tags are needed to identify the image files to download from the Contrail Container Registry during a Contrail Networking installation or upgrade.

The procedures to access the Contrail Container Registry are provided directly by Juniper Networks. The location of the files in the Contrail Container Registry changed for the CN2 software starting in Release 22.4. To obtain access credentials to the registry or if you have any questions about file locations within the registry, send an email to: contrail-registry@juniper.net.

The following table provides the container tag name for the image files for CN2 Release 23.2.

Table 1: Container Tag—Release 23.2

Orchestrator Platform	Container Tag
<ul style="list-style-type: none">• Kubernetes 1.26, 1.25.5, 1.23.9, 1.24.3• Red Hat OpenShift 4.12.13, 4.12.0, 4.10.31, 4.8.39• Amazon EKS v1.24.10-eks-48e63af• RKE 2 v1.27.1+rke2r1	23.2.0.156

Open Issues

IN THIS SECTION

General Routing | 5

General Features | 6

Red Hat OpenShift | 7

CN2 Apstra Integration | 7

CN2 and Kubernetes | 7

Security | 9

CN2 Pipelines | 9

Learn about open issues in this release for CN2.

General Routing

- CN2-3429: When fabric source NAT is enabled in an isolated namespace, traffic flows between pods in isolated namespaces and between pods in isolated and non-isolated namespaces.
Workaround: Do not configure fabric source NAT on an isolated namespace.

General Features

- CN2-3256: cSRX workloads with sub-interfaces are not compatible with CN2.
- CN2-6327: When interface mirroring is enabled with the **juniperheader** option, only egress packets are mirrored.

Workaround: Disable the **juniperheader** option to mirror both egress and ingress packets.

- CN2-5916: When 4 interfaces are configured in a bond interface on an X710 NIC, an mbuf leaf with traffic drop occurs.

Workaround: Limit two interfaces in a bond configuration for an X710 NIC.

- CN2-10346: When restarting a vRouter pod on kernel-mode nodes where vhost0 is installed onto bond interfaces, the bond IP address is assigned to a bond secondary interface instead of a bond primary interface.

Run the following script for the workaround:

```
Bond-patch.txt
text · 982 B

#!/bin/bash

set -x

slave_list=$(ip addr show | grep SLAVE | awk '{ print $2 }' | sed 's://')Revision History
for slave in "${slave_list[@]"; do
    IFS=' '
    bond=$(ip addr show dev ${slave} | grep SLAVE | awk -F'master ' '{print $2}' | awk -F'
' '{print $1}')
    IFS=$'\n'
    route_list=$(ip route show | grep ${slave})
    for route in "${route_list[@]"; do
        echo "route: ${route}"
        new_route=$(echo ${route} | sed "s/${slave}/${bond}/g")
        route_cmd=$(echo "ip route replace ${new_route}" | sed -e 's|["'\''"]|g')
        eval ${route_cmd}
    done
    ipv4=$(ip addr show dev ${slave} | grep 'inet ' | awk '{ print $2 }')
    ipv6=$(ip addr show dev ${slave} | grep 'inet6 ' | awk '{ print $2 }')
    echo "slave: '${slave}', bond: '${bond}', ipv4: '${ipv4}', ipv6: '${ipv6}'"
```

```

if [[ -n "$ipv4" ]]; then
    ip addr del ${ipv4} dev ${slave}
    ip addr add ${ipv4} dev ${bond}
fi
if [[ -n "$ipv6" ]]; then
    ip addr del ${ipv6} dev ${slave}
    ip addr add ${ipv6} dev ${bond}
fi

```

- CN2-13314: The gateway service instance (GSI) does not work with a 4-byte ASN.

Workaround: Use a 2-byte ASN when connecting workloads through the GSI service.

Red Hat OpenShift

- CN2-7787: The KubeVirt deployment in Openshift 4.10 fails intermittently.

See [Red Hat OCPBUGS-2535](#) for a workaround.

- CN2-13011: Red Hat OCP backup and restore fails.

See Red Hat <https://access.redhat.com/solutions/6964756> for a workaround.

CN2 Apstra Integration

- CN2-13607: In a CN2 Apstra deployment, Apstra takes several minutes to create a virtual network under a scaled scenario.

CN2 and Kubernetes

- CN2-4508: Contrail virtualnetwork subnet created through NAD can not have user defined gateway.

Workaround: None.

- CN2-4822: You can not configure BGPaaS objects on nodes that host the Contrail controller and worker nodes on same physical host.

Workaround: None. Production deployments run the Kubernetes worker nodes and controller in different physical hosts.

- CN2-8728: When you deploy CN2 on AWS EC2 instances, running Kubernetes service traffic and Contrail datapath traffic on different interfaces is not supported.

Workaround: Do not deploy Kubernetes and data traffic on the same interface in AWS.

- CN2-10351: KubeVirt v0.58.0 does not support imagePullSecret, required for pulling images from the secure registry: enterprise-hub.juniper.net/contrail-container-prod/.

Following these steps for the workaround:

1. Install Docker.
 2. Create a local insecure registry.
 3. Restart Docker.
 4. Download the required containers. The containers are located at [Release Userspace CNI - dpdk vhostuser interface support Juniper/kubevirt](#). These containers are stored as Assets.
 5. Load the containers.
 6. Tag and push the containers to the new insecure registry.
 7. Download operator.yaml and cr.yaml.
 8. Modify the kubevirt-operator.yaml to use your insecure registry.
- CN2-14895: Pods are being deployed more than the VMI capacity of the nodes.

When a custom pod scheduler is configured with maximum VMI capacity as thresholds, if the pods are scheduled back-to-back in quick succession, it is possible that more pods are deployed than the configured threshold. This is due to the delay in data sync between the node and analytics.

Workaround: Additional pod scheduling on the busy nodes will stop within a few seconds once the VMI data is synced between the nodes and analytics.

- CN2-15530: Packet loss is observed in CN2 flow stickiness when scaling up from one to many pods (non-ECMP to ECMP).

During scale up flow stickiness is applicable only within the ECMP group. Scale up from one to many pods does not maintain flow stickiness.

Workaround: Start with a minimum of 2 workloads and scale up.

- CN2-15461: BFD session is not coming up when healthcheck is associated with 2 BGPaaS objects.

Workaround: In environments where BFD is used with BGPaaS, if firewall policy is configured, ensure that the policy rules allow port 4784 (BFD packets).

Security

- CN2-4642: In CN2, the network policy uses the reserved tags `application` and `namespace`. These tags conflict with Contrail's reserved resources.

Workaround: Do not use the `application` and `namespace` labels to identify the pod and namespace resources.

- CN2-10012: If the network policy has a `deny-all` rule, removing it by updating the policy does not work.

Workaround: Delete the policy and re-add it again.

CN2 Pipelines

- CN2-15876: Tests are triggered when files in a different folder from the one specified in the YAML file directory are committed. The `cn2networkconfig` folder is specified in the `values.yaml` as the directory for commits and files are merged tests expected to be triggered. Argo CD only supports syncing from the path specified in the Helm chart as a part of CN2 pipeline startup.

Workaround: Only commit to the `cn2networkconfig` directory.

- CN2-16034: Auto-created CN2 objects puts Argo out-of-sync after the commit. Creating a NAD starts the virtualRouter and subnets which are flagged as out-of-sync by Argo.

Workaround: Add the `resource.exclusions:` in **`charts/argo-cd/templates/argocd_sa.yaml`**

Workaround added to Helm chart:

```
apiVersion: v1
kind: ConfigMap
metadata:
  namespace: argocd
  labels:
    app.kubernetes.io/name: argocd-cm
    app.kubernetes.io/part-of: argocd
```

```

name: argocd-cm
data:
  resource.exclusions: |
    - apiGroups:
      - "*"
    kinds:
      - VirtualNetwork
    clusters:
      - "*"
  timeout.reconciliation: 2s

```

Resolved Issues

You can research limitations that are resolved with this release at:

[Resolved Issues in CN2 Release 23.2.](#)

Use your Juniper Support login credentials to view the list. If you do not have a Juniper Support account, you can register for one [here](#).

Requesting Technical Support

IN THIS SECTION

- [Self-Help Online Tools and Resources | 11](#)
- [Creating a Service Request with JTAC | 11](#)

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active Juniper Care or Partner Support Services support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the JTAC User Guide located at <https://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf>.

- Product warranties—For product warranty information, visit <https://www.juniper.net/support/warranty/>.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: <https://www.juniper.net/customers/support/>
- Find product documentation: <https://www.juniper.net/documentation/>
- Find solutions and answer questions using our Knowledge Base: <https://kb.juniper.net/>
- Download the latest versions of software and review release notes: <https://www.juniper.net/customers/csc/software/>
- Search technical bulletins for relevant hardware and software notifications: <https://kb.juniper.net/InfoCenter/>
- Join and participate in the Juniper Networks Community Forum: <https://www.juniper.net/company/communities/>
- Create a service request online: <https://supportportal.juniper.net/>

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: <https://entitlementsearch.juniper.net/entitlementsearch/>

Creating a Service Request with JTAC

You can create a service request with JTAC on the Web or by telephone.

- Visit <https://support.juniper.net/support/requesting-support/>
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see <https://support.juniper.net/support/requesting-support/>

Revision History

- 30 June 2023—Revision 6
- 30 March 2023—Revision 5
- 19 December 2022—Revision 4
- 23 September 2022—Revision 3
- 22 June 2022—Revision 2
- 02 May 2022—Revision 1, initial release

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners. Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice. Copyright © 2023 Juniper Networks, Inc. All rights reserved.