
NIAGARA
SUMMIT
2026
SEAMLESS CONNECTIVITY,
POWERFUL INTELLIGENCE
TRIDIUM

This presentation includes content from both Tridium and external speakers. Views expressed by external speakers are their own and do not necessarily reflect the views of Tridium Inc. or its affiliates. Reference to any third-party product, service, customer, or integration does not constitute endorsement by Tridium. Each speaker is responsible for obtaining any required rights or permissions for third-party content included in their portion of this presentation.

Any descriptions of future product direction, intended updates, or new or improved features or functions are for informational purposes only, are subject to change, and do not create any commitment, obligation, or warranty by Tridium. This document contains valuable proprietary and confidential information of Tridium and must not be disclosed to any third party without our written agreement. Content provided by Tridium may not be altered or modified and must remain in the format originally presented.

TRIDIUM

Beyond the VM: Containerized Niagara

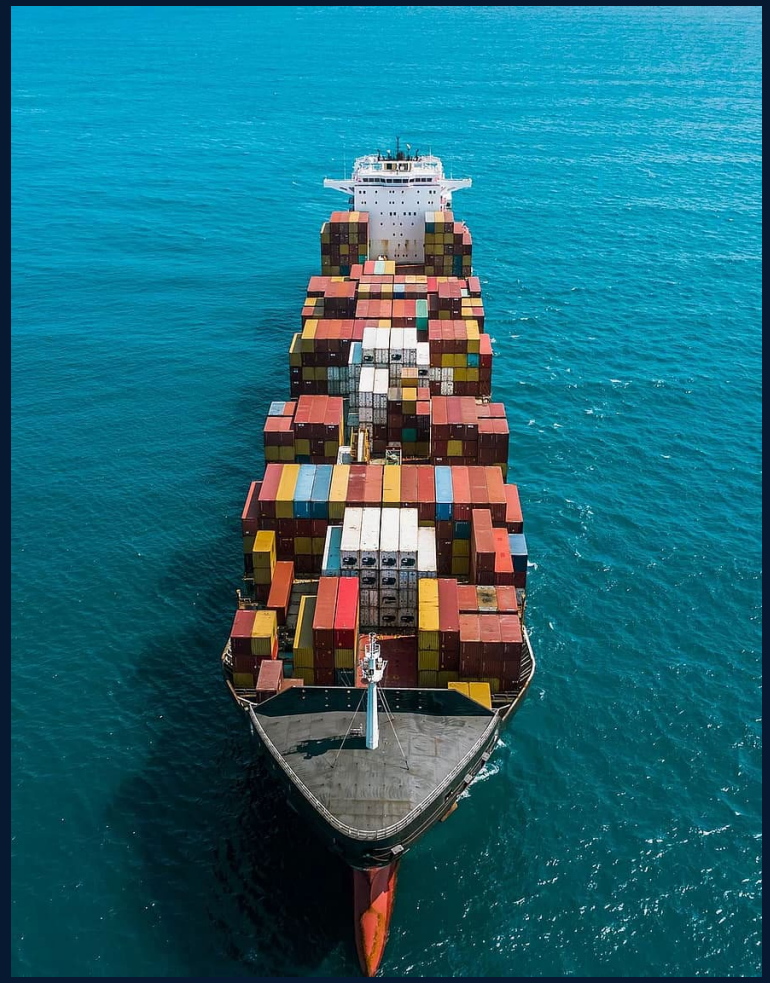
Curtis McKerlie / Tridium
Josh Swanson / Red Hat

NIAGARA SUMMIT 2026

TRIDIUM

What is Containerized Niagara?

What is a container?



- Containers are technologies that allow the packaging and isolation of applications with their entire runtime environment and all of the files necessary to run.

- This makes it easy to move the contained application between environments (dev, test, production, etc.) while retaining full functionality.*

*Source redhat.com

Where can Containerized Niagara be utilized?

EMBEDDED HARDWARE

- For OEMs managing connected devices such as routers and hubs using a container management system, containers can facilitate **easy remote deployment** and **updating of Niagara**

- Portability alternative - embedded deployment whereby a Niagara development partner deploys Niagara containers on its own proprietary or a third-party commercial device

Connected Container Orchestration Platform

CONTAINERIZED PORTABLE NIAGARA (NCC-CPN)

Where can Containerized Niagara be utilized?

CLOUD

- Supervisor deployment in a cloud service

ON PREMISE

- Supervisor deployment on a server device housed locally

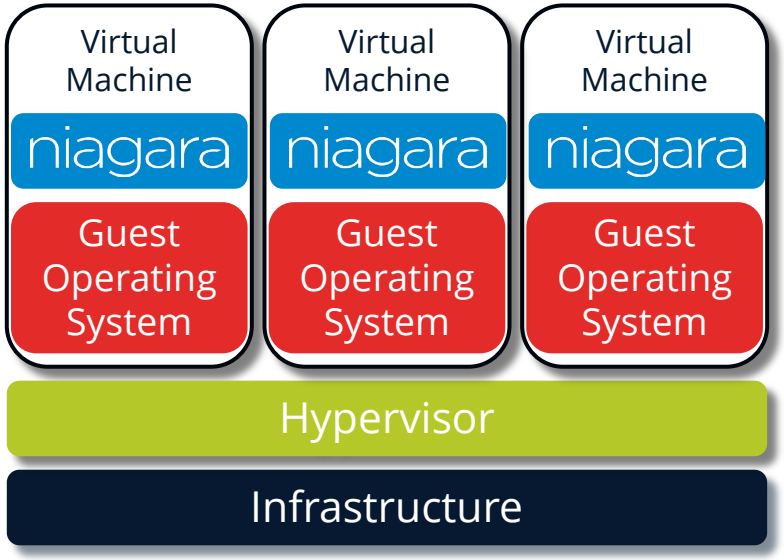
CONTAINERIZED SUPERVISOR (NCC-SUP)

Cloud or Data Centre

Benefits of a Container?

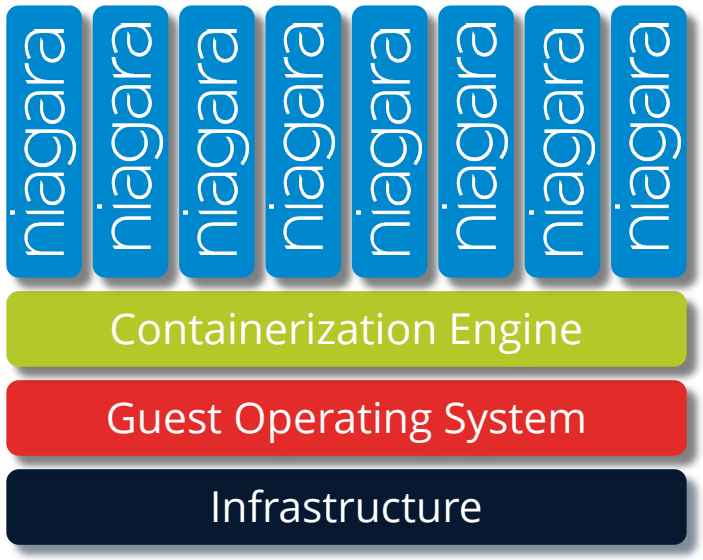
Benefits of containers over VMs?

Virtual Machines



Formerly Popular for Virtualization

Containers



Now Popular for Virtualization

Lightweight:

- Shares the host's OS system kernel
- Does not require an OS per application
- Driving higher server efficiencies
- Reduces server and licensing costs

Secure:

- Applications are safer in containers and provides the strongest default isolation capabilities in the industry

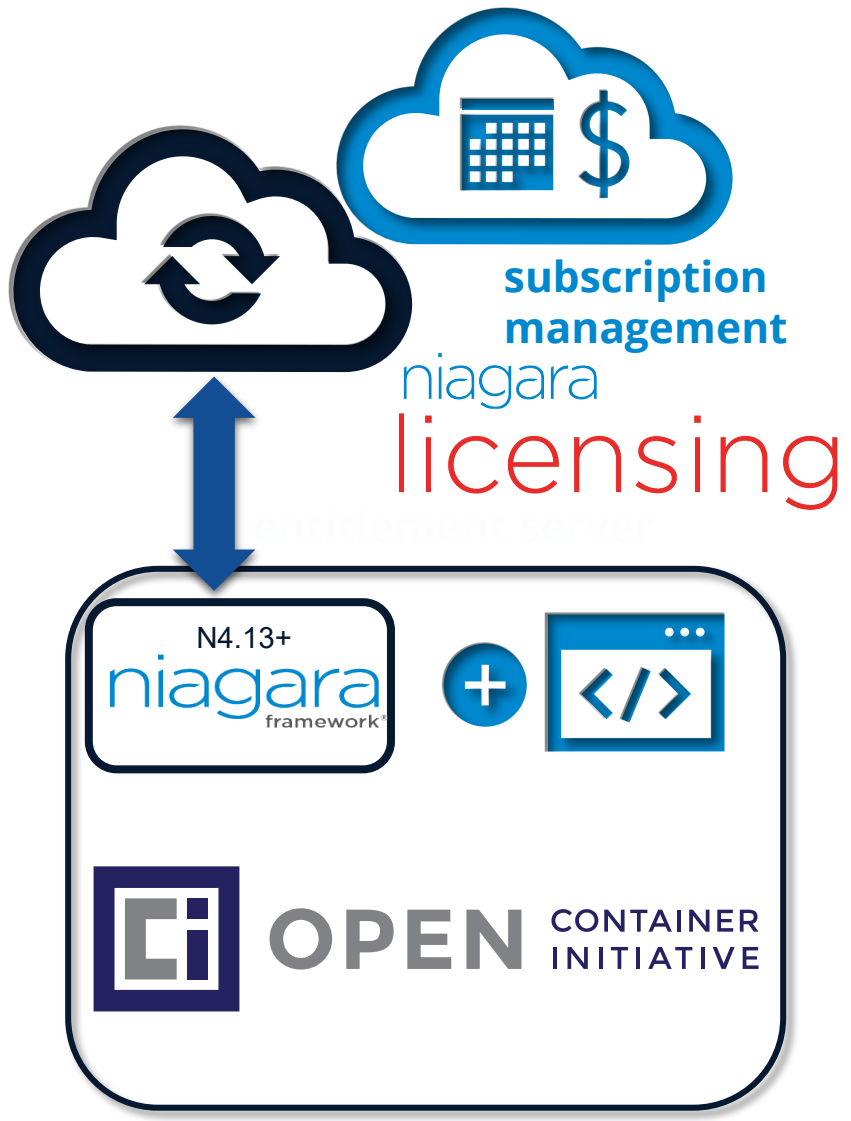
Containers are less resource intensive and more portable than Virtual Machines. Can ease development, deployment and upgrades.

NS2 6

TRIDIUM

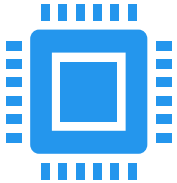
Containerized Niagara

- Niagara Framework supplied in an OCI-compliant container image
- Support for AMD_x86 and ARM64 architecture
- Partner OEMs can add their own applications as required prior to the deployment of their container
- As the container is commissioned, it will contact the Tridium **Entitlement Server** to authorize install, deploy license, and stay activated
- Containerized Niagara was launched with N4.13 as a subscription only service

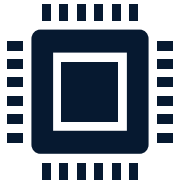


Supported Architectures

- ARM favors simplicity and fast execution of single instructions.
- Uses less power and produces less heat.
- Ideal for embedded devices.



ARM 64



AMD x86

- The x86 processors allow you to perform several activities at the same time from a single instruction.
- Used predominately in servers and PCs.

Authorization Models

- The File Domain Authentication (FDA) version is similar to how JACE is managed today
- Connect with Workbench client using default platform credentials
- Use the wizard to change the default platform credentials and passphrase similar to how a JACE is provisioned out of the box.
- Workbench client can be used to manage the passphrase and platform credentials using the platform administration view

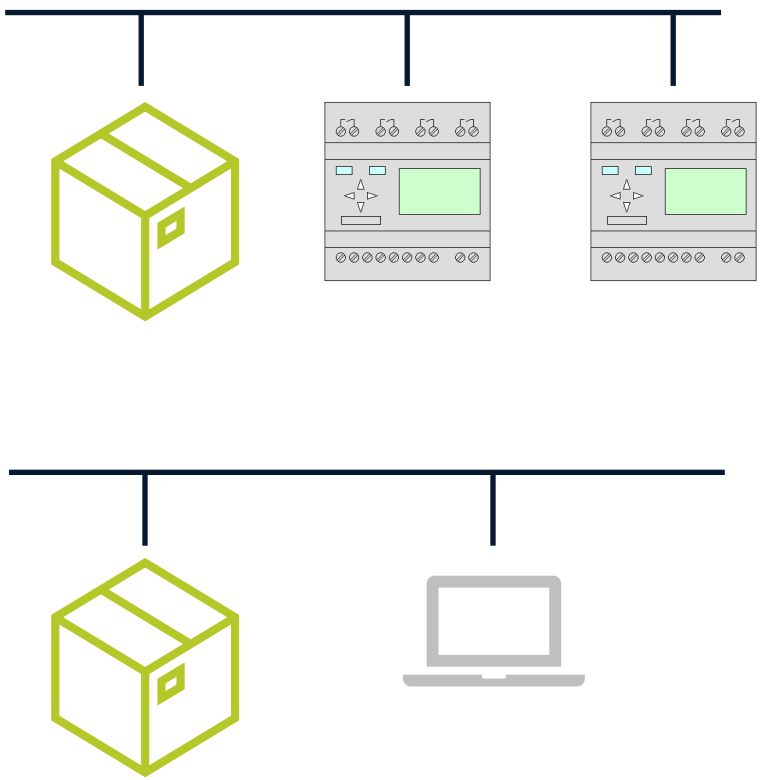
File Domain

Native Domain

- The Native Domain Authentication (NDA) based version requires configuring the passphrase and platform credentials using either environment variables passed into the container runtime or using platform_password and system_passphrase files which are located under a volume which the container has access to
- This uses tools such as Kubernetes Secrets to manage all credentials
- Ideal for large scale deployments of Niagara

Worth noting

- Supervisor Container Models only support IP drivers
- Containers provide headless JAVA runtime
- Remote Workbench environment required to commission
- Browser support for station viewing and configuration

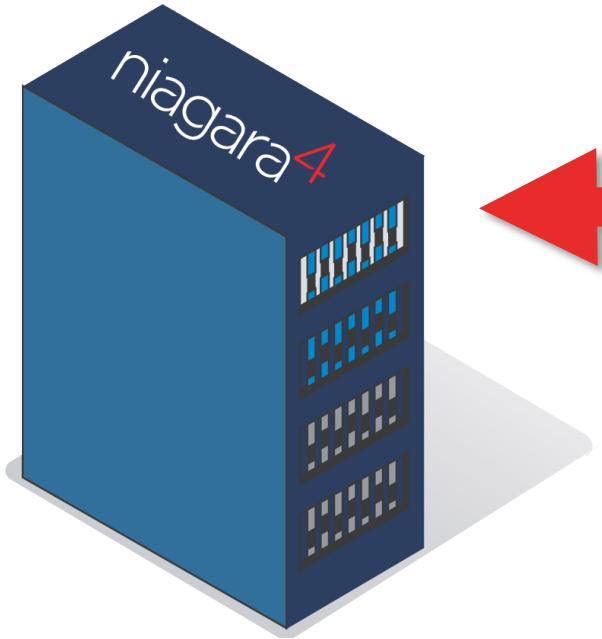


Licensing

Subscription Licensing

SaaS Subscriptions:

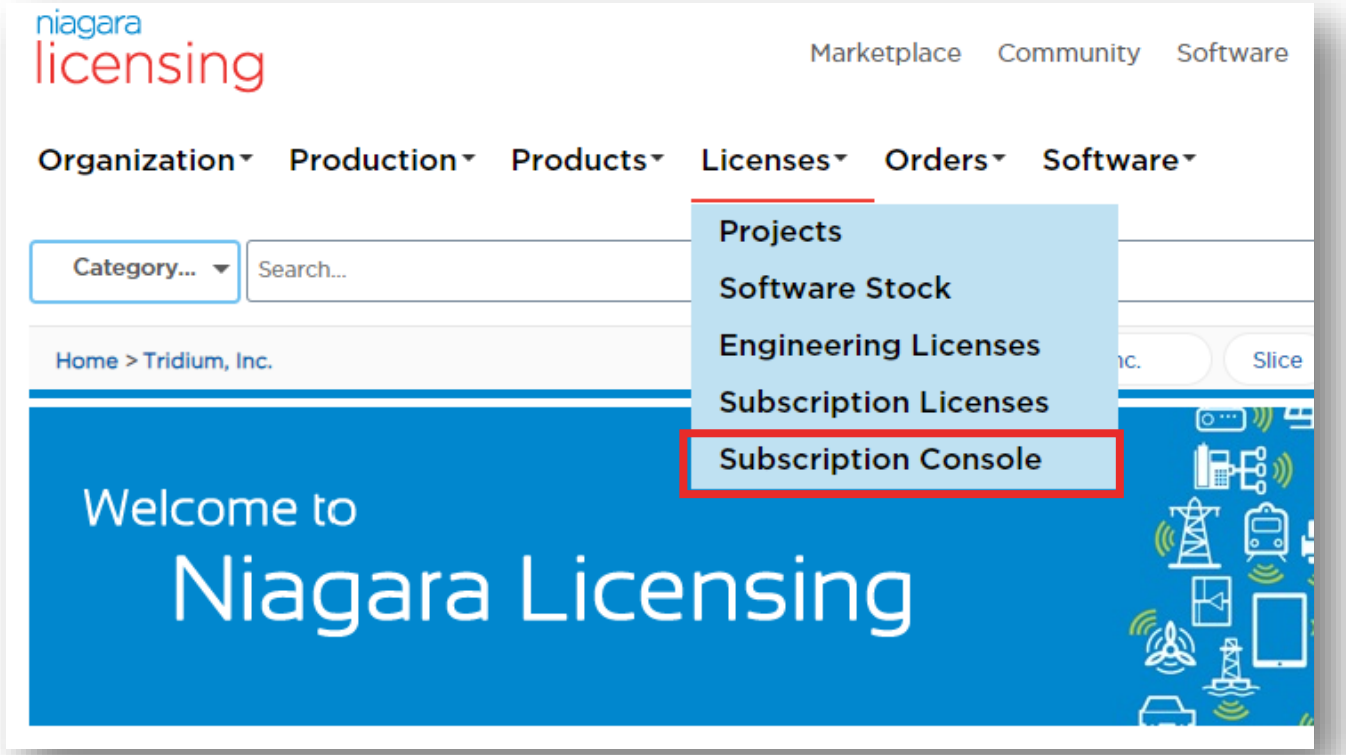
- OPex rather than CAPex
- Less outlay - Less commitment
- SMA included in subscription

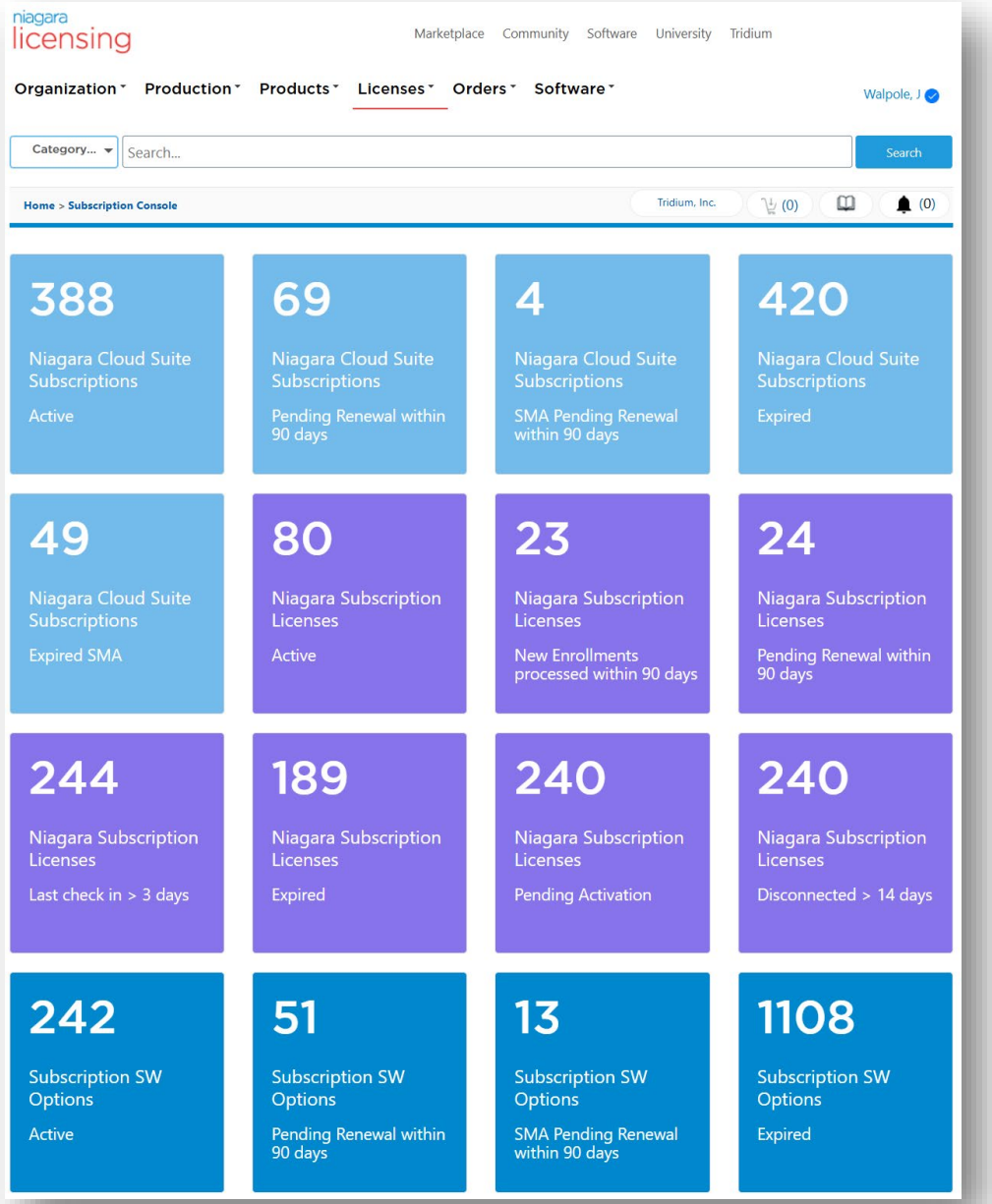


More purchasing flexibility!

- NOTE: A subscription Niagara instance will perform a system exit—that is, it will stop working—if subscription renewal is not paid.

Subscription Licensing



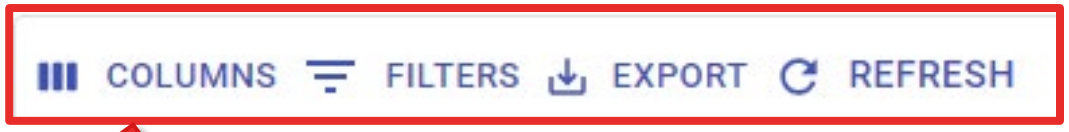


NS26

TRIDIUM

Subscription Licensing

Interactive tables allow selection of appropriate columns, filtering and exporting of data



Home > Subscription Console

Tridium, Inc. (0)

Niagara Subscription Licenses - Active

Niagara Subscription Licenses that are Active and Regularly Checking-In

COLUMNS 1 FILTERS EXPORT REFRESH

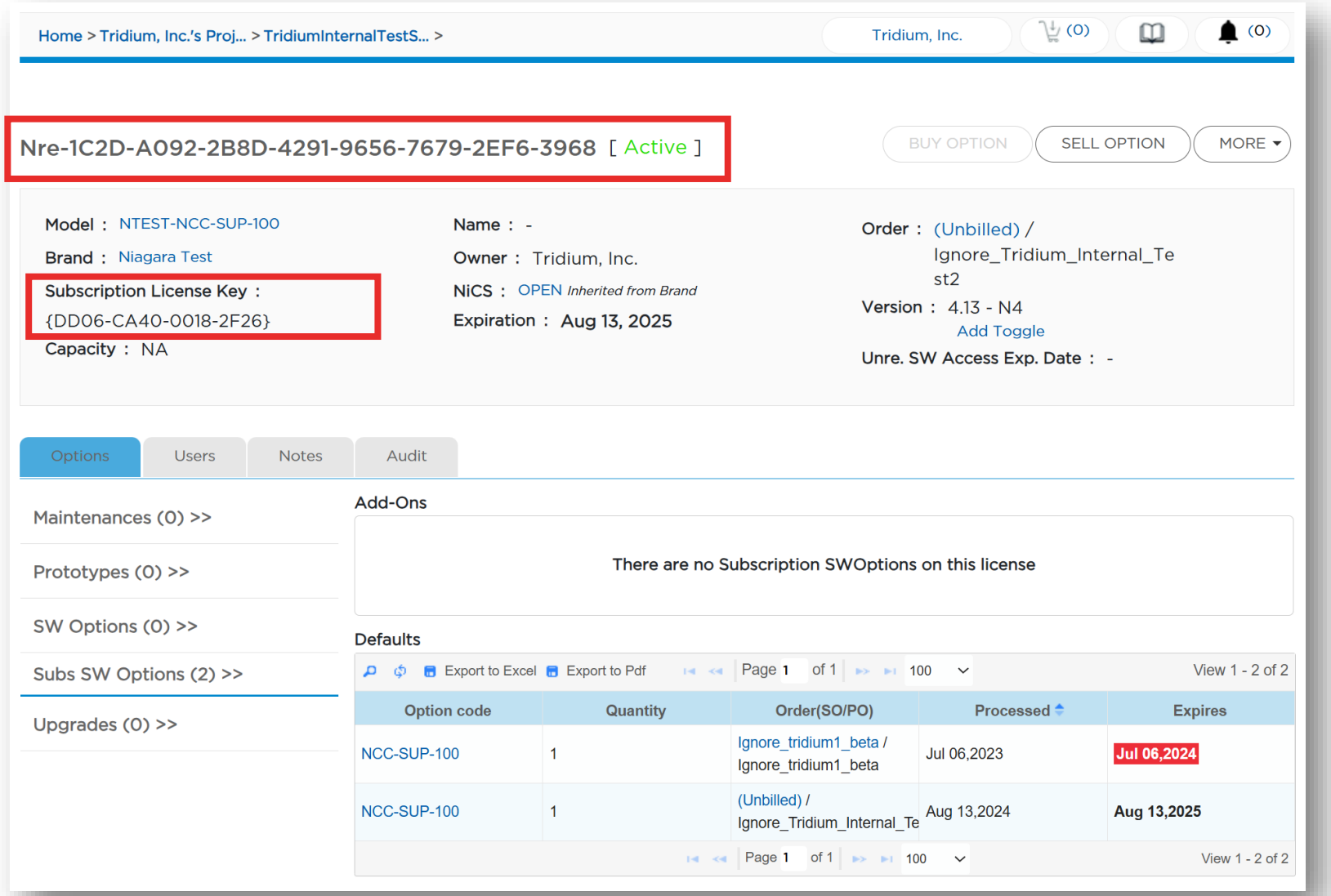
Host Code	Product	Expires	Last Check-In	Project	Subscriber	Project Owner
Nre-1C2D-A092-2B8D-4291-9656-7679-2EF6-3968	NTEST-NCC-SUP-100	14 Aug 2025	Wed Aug 14 2024 16:28:44 GMT+0100	TridiumInternalTestSUP	Tridium, Inc.	Tridium, Inc.
Nre-5EA5-535A-FE16-4E48-8849-31D0-31CA-877B	NTEST-NCC-SUP-100	23 May 2025	Thu Aug 03 2023 11:04:01 GMT+0100	TridiumInternalTestSUP	Tridium, Inc.	Tridium, Inc.
Nre-2F3C-2BCE-D4DC-47F9-ADE4-B640-CF74-7A45	NTEST-NCC-SUP-100	08 Aug 2025	Mon Mar 31 2025 05:53:53 GMT+0100	TRIDIUM UK TRAINING DEPARTMENT	Tridium, Inc.	Tridium, Inc.
Nre-2BE9-15B7-0A15-42A2-BAEC-68F2-281C-B18B	NTEST-J-CPN-0025	23 May 2025	Sun Mar 30 2025 22:51:53 GMT+0100	TridiumInternalTestCPN	Tridium, Inc.	Tridium, Inc.
Nre-8726-E09F-4F5A-4589-BD55-C874-872B-2E17	NTEST-J-CPN-0025	23 May 2025	No Check-In Detected	TridiumInternalTestCPN	Tridium, Inc.	Tridium, Inc.
Nre-E0FF-B67E-A806-4C88-9297-5CC1-6A35-9120	NTEST-NCC-SUP-100	05 Oct 2025	Mon Mar 31 2025 06:53:06 GMT+0100	TRIDIUM UK TRAINING DEPARTMENT	Tridium, Inc.	Tridium, Inc.
Nre-4CC8-5369-AF37-4968-9E66-C8[illegible]B-061D	NTEST-NCC-SUP-100	05 Oct 2025	Mon Mar 31 2025 00:05:39 GMT+0100	TRIDIUM UK TRAINING DEPARTMENT	Tridium, Inc.	Tridium, Inc.

Click through to view each license, Subscriber etc.

NS26

TRIDIUM

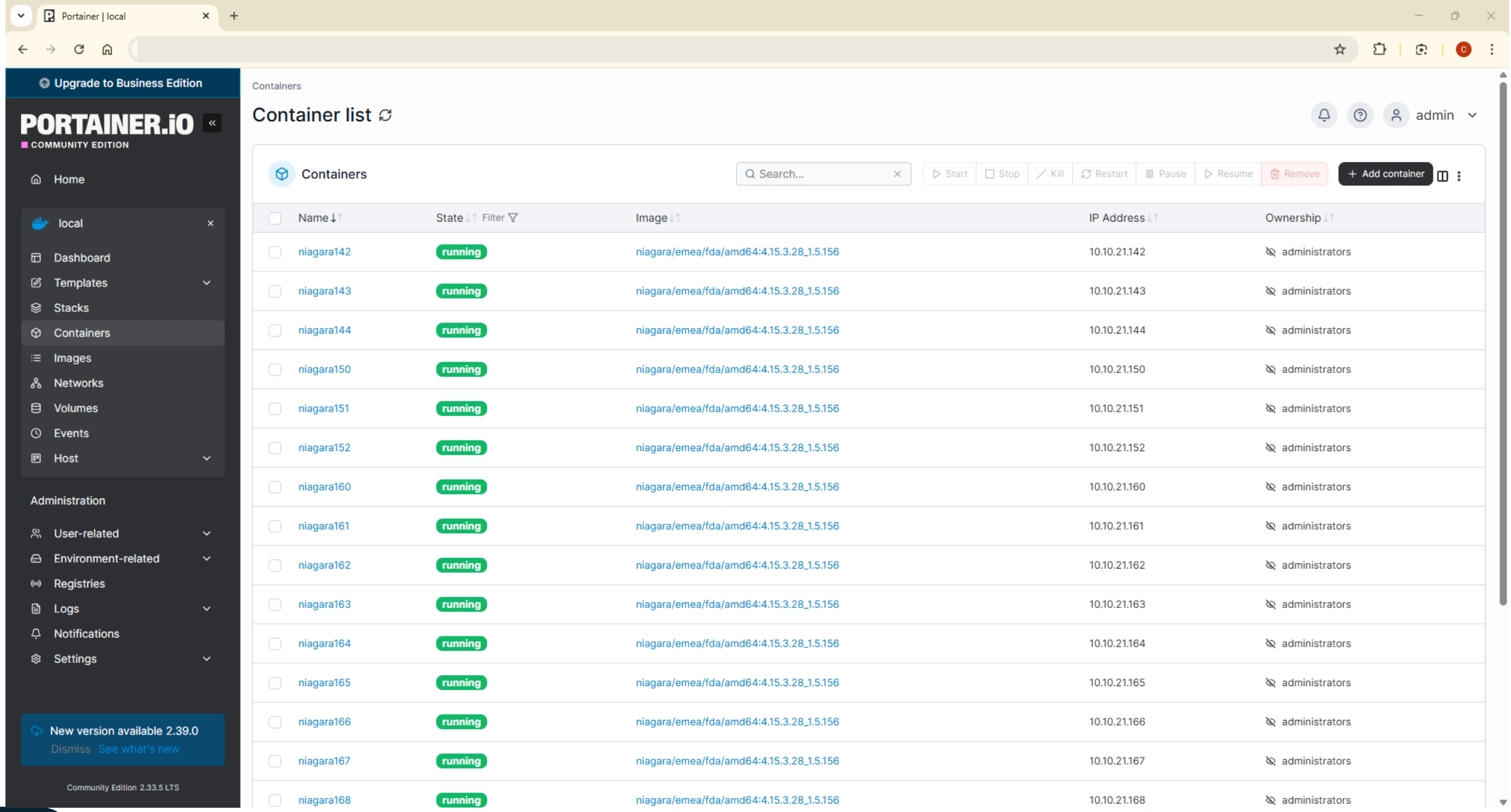
Subscription Licensing



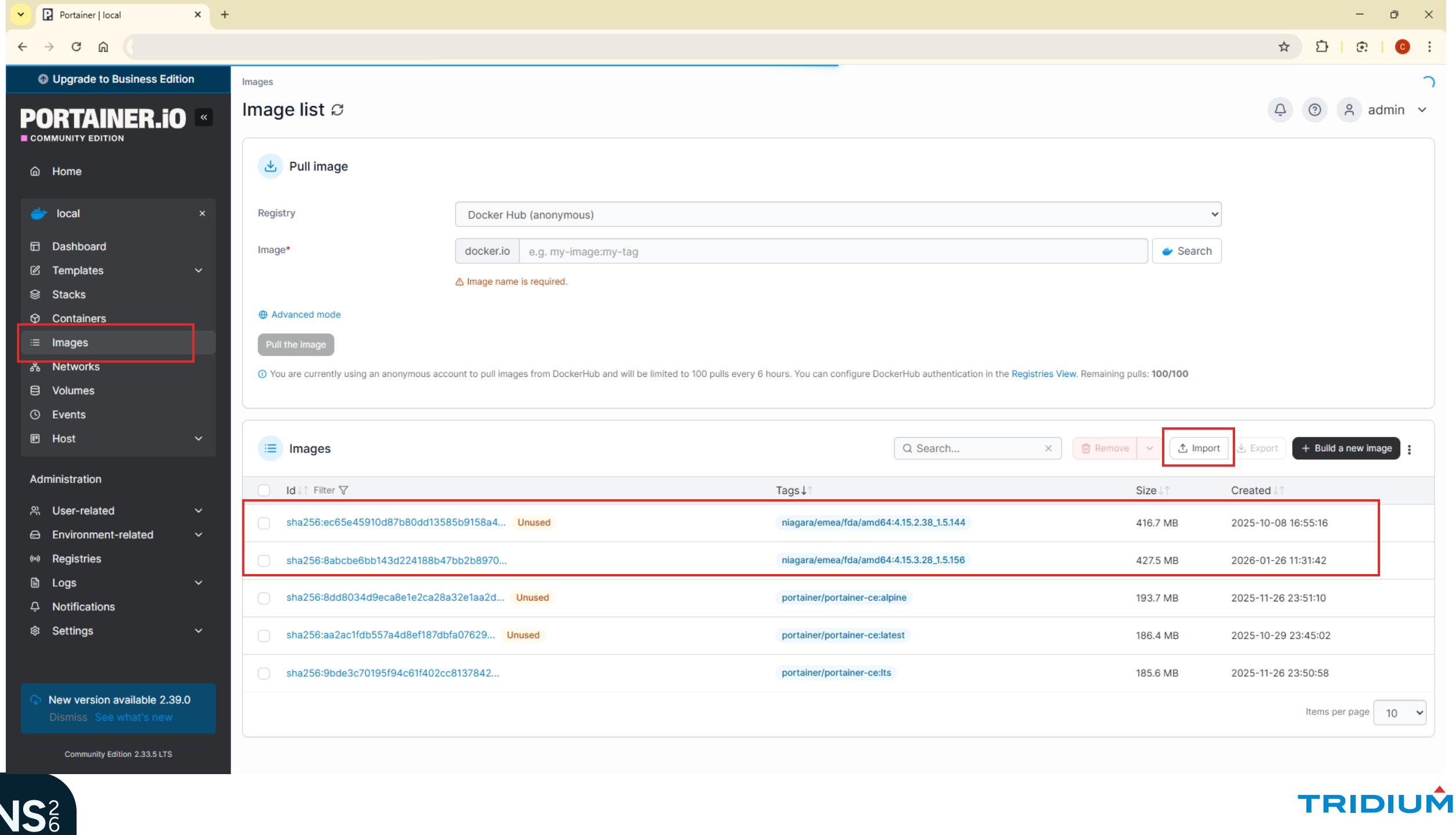
Use Case – EMEA Training Centre

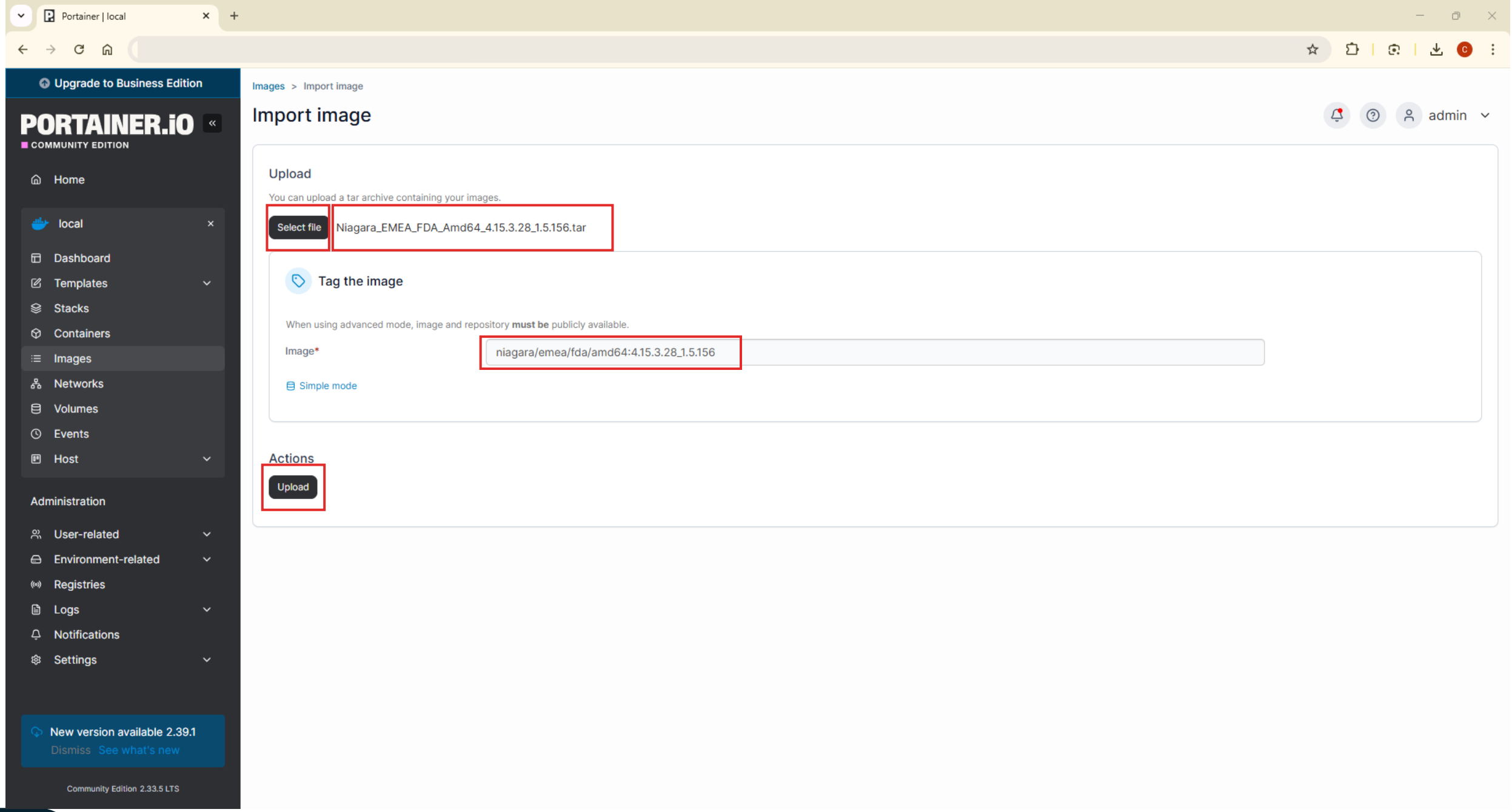
NS26

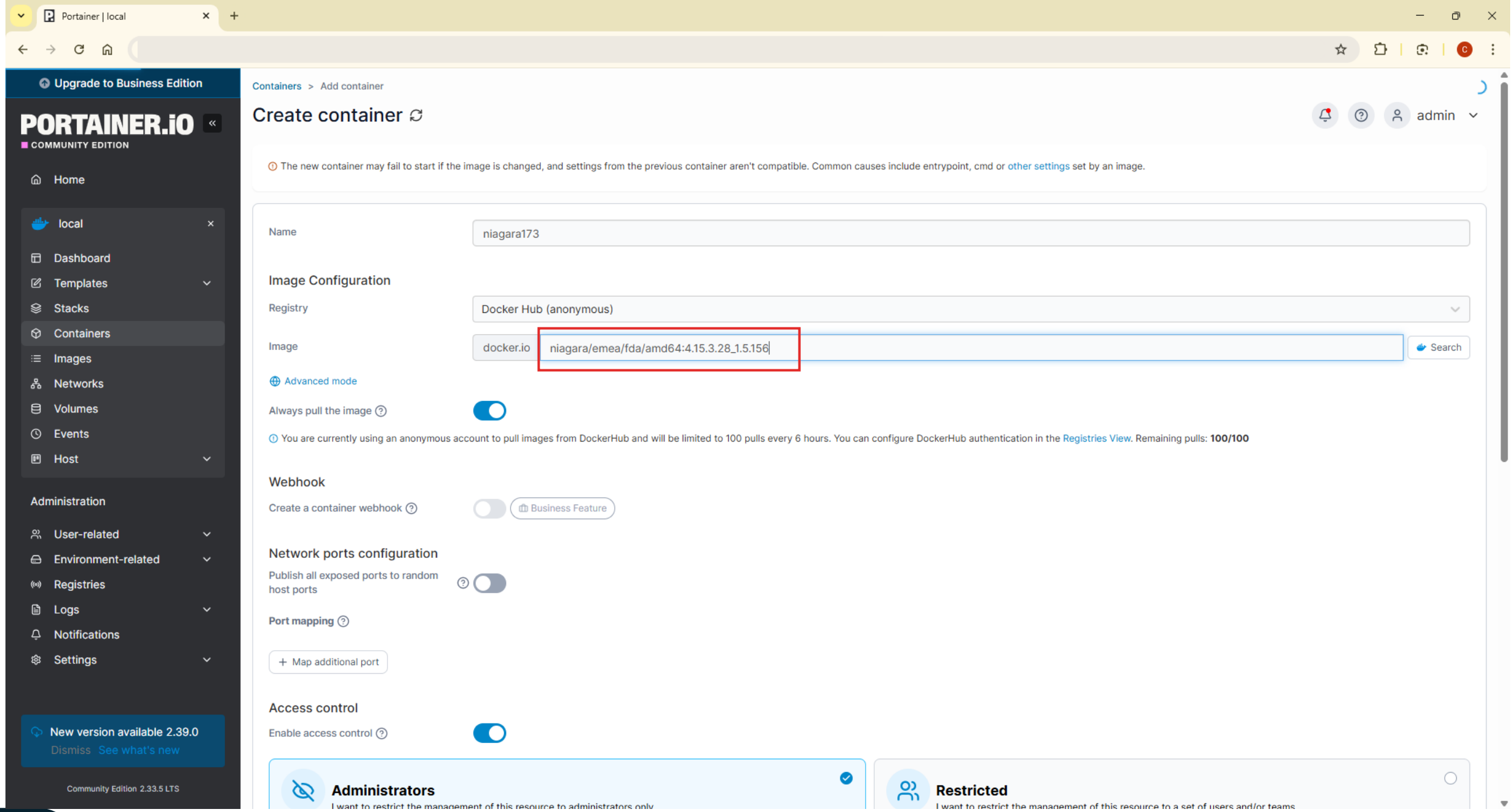
TRIDIUM



* Using Portainer to manage containerized Niagara running in Docker


Portainer | local
Upgrade to Business Edition
PORTAINER.iO
COMMUNITY EDITION
Home
local
Dashboard
Templates
Stacks
Containers
Images
Networks
Volumes
Events
Host
Administration
User-related
Environment-related
Registries
Logs
Notifications
Settings
New version available 2.39.0
Dismiss
See what's new
Community Edition 2.33.5 LTS
Images
Image list
admin
Pull image
Registry
Docker Hub (anonymous)
Image*
docker.io
e.g. my-image:my-tag
Search
Image name is required.
Advanced mode
Pull the image
You are currently using an anonymous account to pull images from DockerHub and will be limited to 100 pulls every 6 hours. You can configure DockerHub authentication in the Registries View. Remaining pulls: 100/100
Images
Search...
Remove
Import
Export
Build a new image
Id
Filter
Tags
Size
Created
sha256:ec65e45910d87b80dd13585b9158a4... Unused
niagara/emea/fda/amd64:4.15.2.38_1.5.144
416.7 MB
2025-10-08 16:55:16
sha256:8abcbe6bb143d224188b47bb2b8970...
niagara/emea/fda/amd64:4.15.3.28_1.5.156
427.5 MB
2026-01-26 11:31:42
sha256:8dd8034d9eca8e1e2ca28a32e1aa2d... Unused
portainer/portainer-ce:alpine
193.7 MB
2025-11-26 23:51:10
sha256:aa2ac1fdb557a4d8ef187dbfa07629... Unused
portainer/portainer-ce:latest
186.4 MB
2025-10-29 23:45:02
sha256:9bde3c70195f94c61f402cc8137842...
portainer/portainer-ce:lts
185.6 MB
2025-11-26 23:50:58
Items per page
10


Portainer | local
Upgrade to Business Edition
PORTAINER.IO
COMMUNITY EDITION
Home
local
Dashboard
Templates
Stacks
Containers
Images
Networks
Volumes
Events
Host
Administration
User-related
Environment-related
Registries
Logs
Notifications
Settings
New version available 2.39.1
Dismiss
See what's new
Community Edition 2.33.5 LTS
Images > Import image
Import image
admin
Upload
You can upload a tar archive containing your images.
Select file
Niagara_EMEA_FDA_Amd64_4.15.3.28_1.5.156.tar
Tag the image
When using advanced mode, image and repository must be publicly available.
Image*
niagara/emea/fda/amd64:4.15.3.28_1.5.156
Simple mode
Actions
Upload


Portainer | local
Upgrade to Business Edition
PORTAINER.IO
COMMUNITY EDITION
Home
local
Dashboard
Templates
Stacks
Containers
Images
Networks
Volumes
Events
Host
Administration
User-related
Environment-related
Registries
Logs
Notifications
Settings
New version available 2.39.0
Dismiss See what's new
Community Edition 2.33.5 LTS
Containers > Add container
Create container
admin
The new container may fail to start if the image is changed, and settings from the previous container aren't compatible. Common causes include entrypoint, cmd or other settings set by an image.
Name
niagara173
Image Configuration
Registry
Docker Hub (anonymous)
Image
docker.io
niagara/emea/fda/amd64:4.15.3.28_1.5.156
Search
Advanced mode
Always pull the image
You are currently using an anonymous account to pull images from DockerHub and will be limited to 100 pulls every 6 hours. You can configure DockerHub authentication in the Registries View. Remaining pulls: 100/100
Webhook
Create a container webhook
Business Feature
Network ports configuration
Publish all exposed ports to random host ports
Port mapping
Map additional port
Access control
Enable access control
Administrators
I want to restrict the management of this resource to administrators only
Restricted
I want to restrict the management of this resource to a set of users and/or teams

High Availability & Fault Tolerance

High Availability vs Fault Tolerance

High Availability

- **Goal:** Maximize uptime, minimize service interruptions
- **Method:** Detect failure and switches to backup component (failover)
- **Impact:** Users may notice a brief pause during switch
- **Cost:** Generally lower, easier to implement than FT
- **Example:** A load balancer redirecting traffic from a failed server to a working one

Key Difference Summary

- **Downtime:** HA accepts some downtime; FT strives for zero
- **Recovery:** HA recovers after a failure; FT operates through a failure
- **Use Case:** HA suitable for web services; FT is necessary for critical infrastructure such as banking, healthcare and data centers

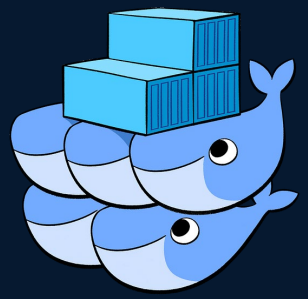
Fault Tolerance

- **Goal:** Zero downtime; continuous operation
- **Method:** Redundant components run in parallel, taking over instantly
- **Impact:** Users are unlikely to notice a disruption
- **Cost:** High, due to complex, mirrored infrastructure
- **Example:** A critical database using synchronous replication to a hot standby, ensuring no data loss or interruption

High Availability vs Fault Tolerance

High Availability Solutions

- **Docker Swarm**



- **Kubernetes**

Fault Tolerance Solution

- **Penguin Solutions' Stratus**

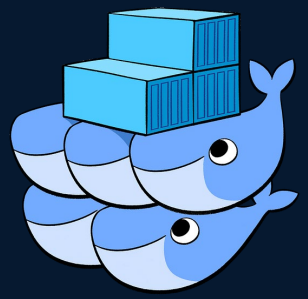
Stratus
A PENGUIN SOLUTIONS® BRAND

The following comparison is provided for informational purposes to help customers evaluate technology options for use with Niagara. All information is based on publicly available documentation as of Apr 2026. Tridium does not endorse, certify, or guarantee the performance of any third-party solution. Customers should verify product capabilities directly with each vendor. Feature availability may vary by version, configuration, and deployment.

Overview

High Availability Solutions

- **Docker Swarm**



Docker Swarm is an open-source platform by Docker for container orchestration, which enables multiple Docker instances to operate as a single virtual host. A Docker Swarm cluster has three key components:

Key Features:

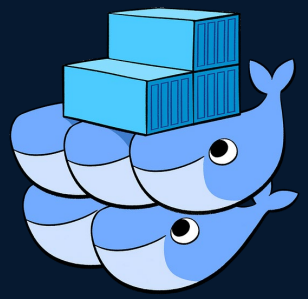
- **Nodes**: These are the Docker engine instances that manage the cluster and the containers for services and tasks.
- **Services and tasks**: The applications and workloads managed within the cluster.
- **Load balancers**: They distribute requests across nodes.

For high availability, Docker Swarm is designed so that if a node fails, its services are reassigned to other nodes. A highly available Swarm cluster requires at least three manager nodes to avoid interruptions during incidents.

Comparison Details

High Availability Solutions

- **Docker Swarm**



Considerations for Docker Swarm:

- **Ease of Installation and Use**: Designed to be lightweight and accessible, which may suite users newer to container orchestration.
- **Quick to Learn**: Generally, requires less time to understand relative to other orchestration tools.
- **Automated Load Balancing**: Supports automated load balancing among Docker containers.
- **CLI Integration**: Works with the existing Docker CLI, avoiding the need for a new interface or configuration changes.
- **Integration**: Integrates with Docker tools like Docker Compose.

Trade-offs to consider with Docker Swarm:

- **Scope of Functionality**: As a lighter-weight tool relying on the Docker API, it may offer fewer capabilities than Kubernetes depending on deployment requirements.
- **Automation Scope**: Automation features may be more limited relative to Kubernetes, based on specific use-case needs.

Overview

High Availability Solutions

- **Kubernetes**

Kubernetes (K8s) is an open-source platform for automating the deployment, scaling, and management of containerized applications. Developed by Google, it is known for its flexibility and reliability.

Key Features:

- **Automatic Scaling**: Adjusts resource allocation based on demand.
- **Load Balancing**: Efficiently distributes traffic for high availability.
- **Self-Healing**: Restarts or replaces failed containers to ensure uptime.
- **Declarative Configuration**: Uses YAML or JSON files for state management.
- **Service Discovery**: Enables communication between containers and services.
- **Storage Orchestration**: Supports various storage systems.
- **Rolling Updates**: Allows updates without downtime.
- **Multi-Cloud Support**: Functions across multiple environments, including on-premises and cloud.

Kubernetes is essential for managing complex containerized applications and microservices architectures.

Comparison Details

High Availability Solutions

- **Kubernetes**

Considerations for Kubernetes

Kubernetes offers a wide range of benefits to teams who need a robust container orchestration tool:

- It has a large open-source community, backed by Google.
- It offer compatibility across various operating systems.
- It is capable of managing large architectures and handling complex workloads effectively.
- It includes automation and self-healing capabilities, enabling automatic scaling.
- It provides built-in monitoring features and supports numerous integrations.
- Is available on major cloud providers, including Google, Azure, and AWS.

Due to its extensive community-support and capability to manage complex deployment scenarios, Kubernetes is frequently a top choice for enterprise development teams managing microservice-based applications.

Comparison Details

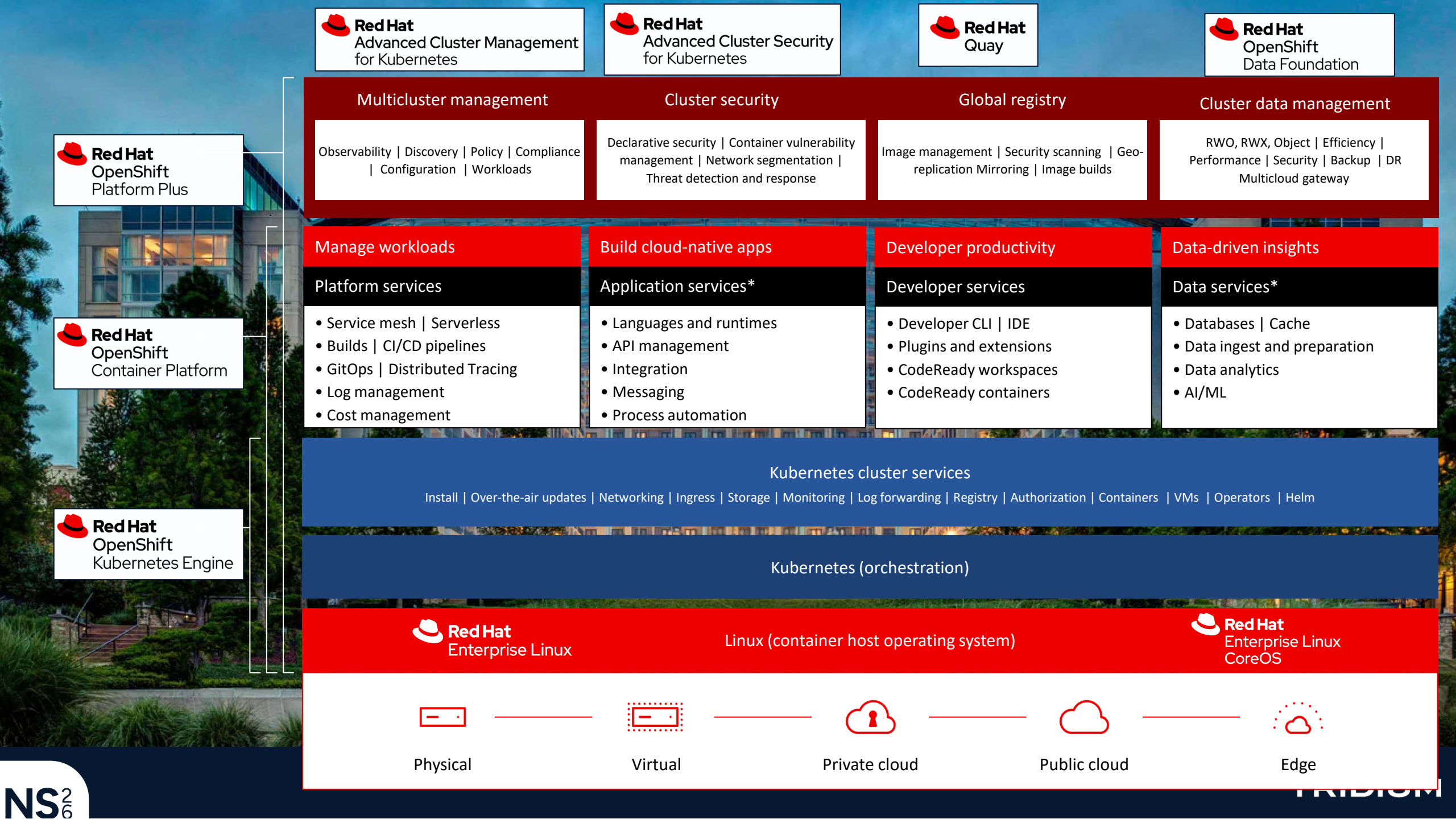
High Availability Solutions

- **Kubernetes**

Trade-offs to consider with Kubernetes

- The installation process can be complex, and users may face a steep learning curve.
- It necessitates the installation of multiple command-line interface (CLI) tools, each of which require additional learning.
- Transitioning from simpler orchestration tools like Docker to Kubernetes could pose challenges and management difficulties.

In certain cases, the complexity of Kubernetes may lead to reduced productivity compared to other, more straightforward orchestration options.


Red Hat Advanced Cluster Management for Kubernetes
Red Hat Advanced Cluster Security for Kubernetes
Red Hat Quay
Red Hat OpenShift Data Foundation
Red Hat OpenShift Platform Plus
Red Hat OpenShift Container Platform
Red Hat OpenShift Kubernetes Engine
Multicluster management
Observability | Discovery | Policy | Compliance | Configuration | Workloads
Cluster security
Declarative security | Container vulnerability management | Network segmentation | Threat detection and response
Global registry
Image management | Security scanning | Geo-replication Mirroring | Image builds
Cluster data management
RWO, RWX, Object | Efficiency | Performance | Security | Backup | DR Multicloud gateway
Manage workloads
Platform services
• Service mesh | Serverless
• Builds | CI/CD pipelines
• GitOps | Distributed Tracing
• Log management
• Cost management
Build cloud-native apps
Application services*
• Languages and runtimes
• API management
• Integration
• Messaging
• Process automation
Developer productivity
Developer services
• Developer CLI | IDE
• Plugins and extensions
• CodeReady workspaces
• CodeReady containers
Data-driven insights
Data services*
• Databases | Cache
• Data ingest and preparation
• Data analytics
• AI/ML
Kubernetes cluster services
Install | Over-the-air updates | Networking | Ingress | Storage | Monitoring | Log forwarding | Registry | Authorization | Containers | VMs | Operators | Helm
Kubernetes (orchestration)
Red Hat Enterprise Linux
Linux (container host operating system)
Red Hat Enterprise Linux CoreOS
Physical
Virtual
Private cloud
Public cloud
Edge

Overview

Stratus everRun is a fault-tolerant software solution that ensures continuous availability for essential applications, guaranteeing zero downtime and preventing data loss by duplicating workloads across two standard x86 servers.

Key Features:

- **Easy Installation**: No changes needed to applications for integration.
- **Automatic Recovery**: Quickly recovers from hardware or software failures.
- **Cost-Effective**: Provides high-availability protection at an affordable price.

Stratus everRun is offered in two formats: a software-only option that leverages existing hardware and a comprehensive package that includes both hardware and software for a seamless, turnkey setup. This flexibility allows users to choose the best fit for their needs and infrastructure.

Fault Tolerance Solution

- **Penguin Solutions' Stratus**

Comparison Details

Considerations for Stratus everRun:

- **SplitSite Functionality:** Provides geographic redundancy through synchronous replication across sites located up to 6 miles apart, thereby ensuring uptime during site-wide disasters.
- **Proactive Monitoring:** Includes a web-based console that monitors system health, offering diagnostic and alerts aimed at preventing issues.
- **Maintenance Without Downtime:** Supports a "work-on" mode, allowing maintenance or upgrades on one server while keeping the workload operational on the second node.

These capabilities are important considerations when evaluating fault-tolerant solutions in various contexts.

Fault Tolerance Solution

- **Penguin Solutions' Stratus**

Comparison Details

Trades-offs to consider for Stratus everRun:

Two-Node Constraint: everRun is inherently limited to a two-node configuration and does not support scaling by adding more physical servers to the cluster.

High Licensing Cost: Solutions that provide continuous availability typically incur higher license fees compared to standard high-availability (HA) setups.

Resource Intensive: Running in Fault Tolerant (FT) mode requires applications to run in lock-step, which demands significant network bandwidth for synchronization (availability links).

Potential Performance Issues: Large virtual disk volumes (2TB or greater) may suffer from performance degradation if fragmentation occurs over time.

Hardware Restrictions: Although it can run on off-the-shelf hardware, it requires specific processor types (Intel) and is primarily designed for x86 architecture, limiting flexibility compared to other solutions.

These trade-offs should be carefully considered when evaluating fault-tolerant systems.

Fault Tolerance Solution

- **Penguin Solutions' Stratus**


Red Hat Enterprise Linux

Red Hat Enterprise Linux

Ensured stability and deployment flexibility

Image mode for RHEL

Thin, optimized OS reduces attack surface

Easily create purpose-built OS images optimized for the architectural challenges inherent at edge and without unnecessary packages.

Remote device update mirroring

Staged and applied image updates occur at the next reboot or power cycle, ensuring minimal downtime.

Efficient over-the-air updates

Updates transfer significantly less data and are ideal for remote sites with limited or intermittent connectivity. Make it easier to keep up to date with the latest security fixes.

Intelligent rollbacks

Application specific health checks detect conflicts and automatically revert an operating system (OS) update, preventing downtime.

NS26

TRIDIUM

Image mode for RHEL

A container-native workflow for the life cycle of a system

Build

A *bootc* base image & container file is all that's needed to describe a system, applications, and dependencies. Use your existing container tools or pipelines to quickly create and test images.

Deploy

Easily convert to a VM/cloud image or deploy on bare metal using RHEL's installer. The container image includes full hardware drivers, but not cloud agents by default.

Manage

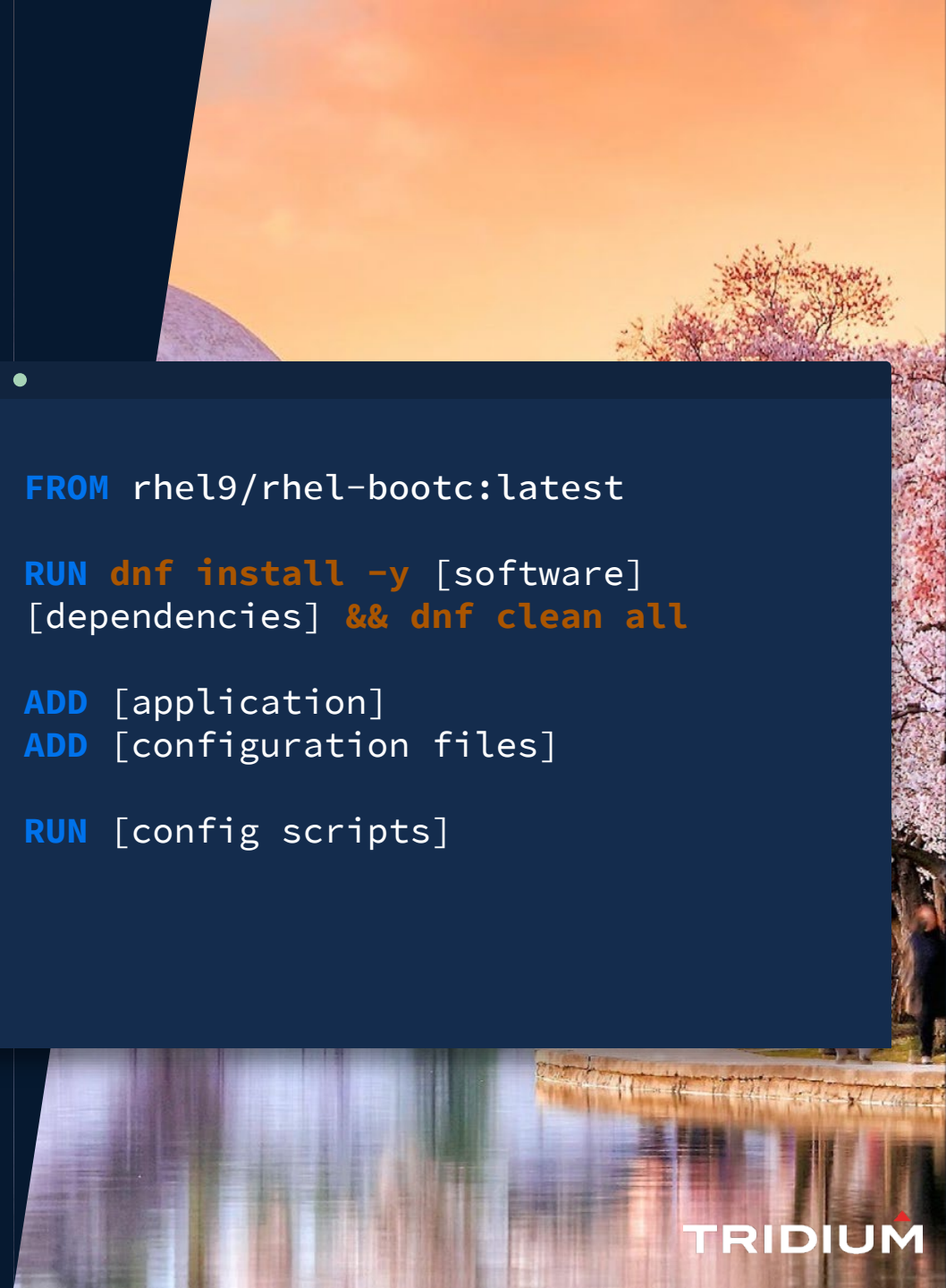
Designed for modern GitOps & CI/CD driven environments. Systems will auto-update from the container registry by default. More advanced control and automation is available via custom rollouts (e.g. Ansible). Intelligence via Insights and on-prem content curation via Satellite are planned for the future.

```
FROM rhel9/rhel-bootc:latest

RUN dnf install -y [software]
[dependencies] && dnf clean all

ADD [application]
ADD [configuration files]

RUN [config scripts]
```



NS26

TRIDIUM

Red Hat Edge Manager

Comprehensive fleet management

Simple

Intuitive edge operations

Bridge the IT skills gap at the edge with a user-friendly environment designed for ease of use and simplified management.

Opinionated management options

Deployment models, with on-premises management offering comparable or superior value to cloud alternatives.

Scalable

Policy-driven deployment

Implement a desired-state configuration model for both applications and infrastructure, enabling consistent and scalable deployments, maximizing operational efficiency.

Resilient pull-mode management

Use a robust agent-based architecture for scalable device management, maintaining connectivity and control even in challenging network conditions, without complex network configurations.

Optimized for edge

Reliable ifecycle management

Applications and OS management with secure onboarding, upgrades policies, change error budgets of applications and OS.

Hardened security

Ensure devices are secure up-to-date and adhering to standards, all backed by secure and robust agent communication. Ensuring a consistent and highly secure posture for devices with limited connectivity.

Proactive device insights

Monitor critical device compute, memory and disk resources. Capture essential metrics and logs for effective issue remediation and operational awareness.

NS26

TRIDIUM

What is fleet management?

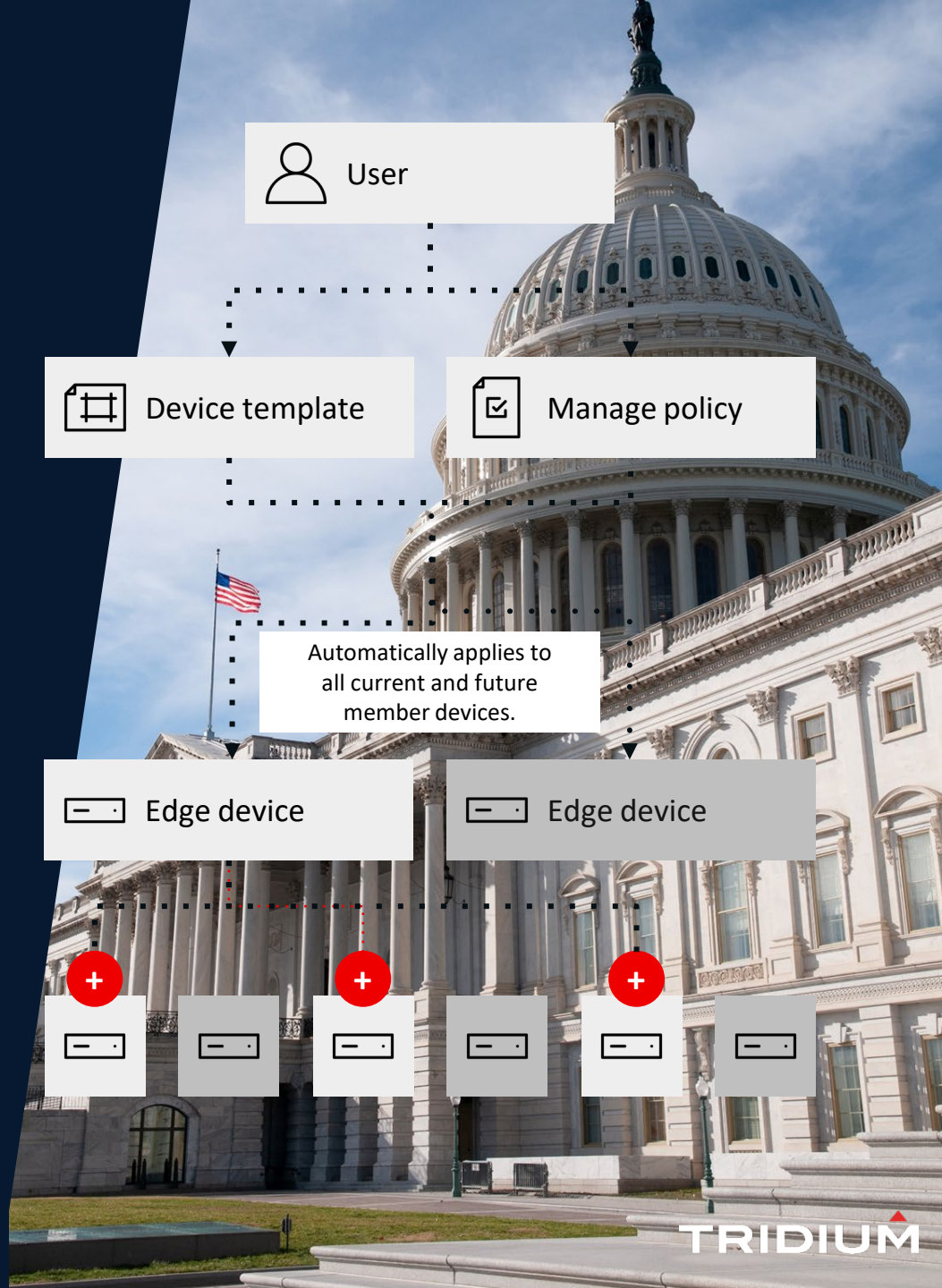
A fleet is a collection of 1 or more edge devices that adhere to fleet specifications

What does a fleet specify?

- Operating system image
- Operating system configuration
- Applications

What is the business value?

- Helps eliminate configuration drift
- Lowers operating expenses



NS26

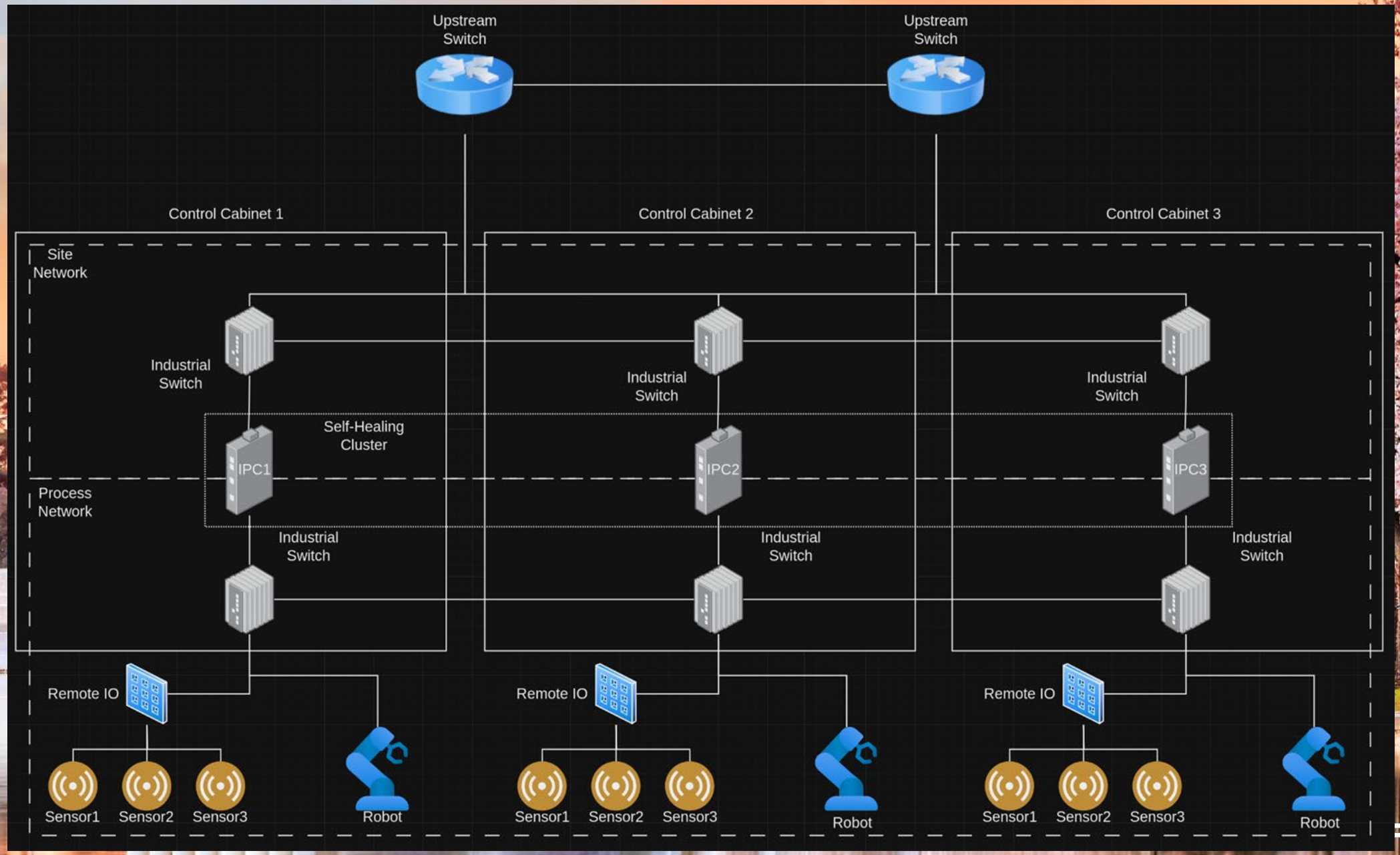
TRIDIUM

What is Pacemaker?

Pacemaker is a high availability Cluster Resource Manager (CRM) that can be used to manage resources, and ensure that they remain available in the event of a node failure

- Detection and recovery of machine and application-level failures
- Supports practically any redundancy configuration
- Supports both quorate and resource-driven clusters
- Configurable strategies for dealing with quorum loss (when multiple machines fail)
- Supports application startup/shutdown ordering, regardless machine(s) the applications are on
- Supports applications that must/must-not run on the same machine
- Supports applications which need to be active on multiple machines
- Supports applications with multiple modes (such as master/slave)

Pacemaker In Action



Any Questions?

NIAGARA SUMMIT 2026

TRIDIUM

Thank you for your time

curtis.mckerlie@tridium.com
joshswanson@redhat.com

TRIDIUM