



**Connect**

# Serverless in Edge Computing

Serverless with Podman and RHEL for Edge

Luis Arizmendi

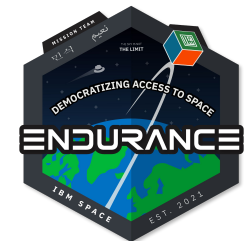
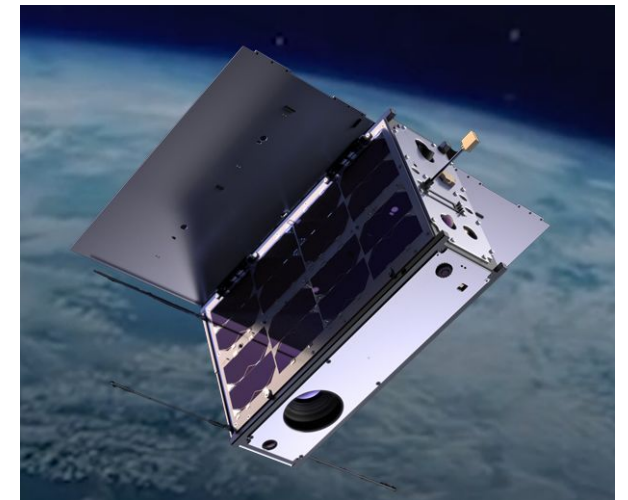
Principal Specialist Solution Architect

José Ángel de Bustos

Senior Specialist Solution Architect

# This is “Edge”!

Edge Computing “What” and “Why” use case examples

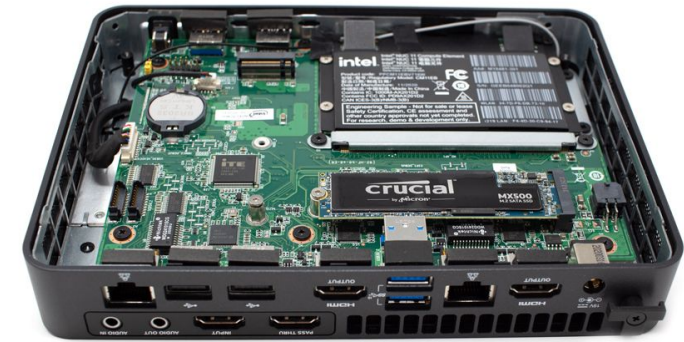


# The feature-rich vs small-footprint trade off

The right balance between functionality and hardware footprint





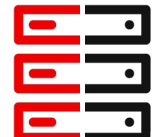
**VMS**  
**Containers**  
**Automated Ops**

**Serverless**



# Red Hat has already an Edge Computing portfolio

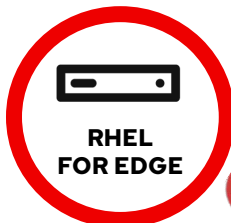



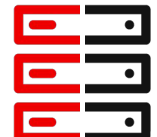
## Red Hat Edge Computing products

	 RHEL FOR EDGE	 RHEL FOR EDGE + MICROSHIFT *	 OPENSIFT SINGLE NODE	 OPENSIFT REMOTE WORKER NODES	 OPENSIFT 3-NODE
Definition	A small footprint deployment on hardware which cannot handle full OpenShift cluster. Key building blocks are <b>RHEL</b> and a <b>container runtime</b> .		<b>OpenShift</b> deployment on a single box ( <b>C</b> ontrol Plane + <b>W</b> orker) with fair amount of resources.	<b>OpenShift C</b> ontrol Plane reside in a central location, whereas <b>W</b> orkers are distributed remotely at the edge sites sharing the control plane.	<b>OpenShift C</b> ontrol Plane and <b>W</b> orkers reside on the same node. HA setup with 3 servers.
Examples	<div>Fits in where OCP in a single node doesn't fit (e.g. HW restrictions)</div> <div>IoT / data collection gateways</div> <div>uCPE (customer premise equipment)</div>	<div>In-vehicle field operations</div> <div>Telco 5G sparsely populated areas</div> <div>In-field single server operations</div> <div>Disconnected environments</div>	<div>Telco 5G far edge - RAN</div> <div>IoT / data collection gateways</div>	<div>Telco 5G near edge &amp; MEC</div> <div>Edge AI &amp; Data pipelining</div> <div>Smart manufacturing</div> <div>Remote office</div> <div>Disconnected clusters</div>	

\* Roadmap, not a product yet

# Red Hat has already an Edge Computing portfolio

## Red Hat Edge Computing products

	 RHEL FOR EDGE	 RHEL FOR EDGE + OSH	 OPENSIFT SINGLE NODE	 OPENSIFT REMOTE WORKER NODES	 OPENSIFT 3-NODE	
Definition	A small footprint deployment on hardware which cannot handle a full OpenShift cluster. Key building blocks are <b>RHEL</b> and a <b>container runtime</b> .		Deployment on a single box (Control Plane + <b>Worker</b> ) with fair amount of resources.		<b>OpenShift C</b> ontrol Plane reside in a central location, whereas <b>W</b> orkers are distributed remotely at the edge sites sharing the control plane.	<b>OpenShift C</b> ontrol Plane and <b>W</b> orkers reside on the same node. HA setup with 3 servers.
Examples	<div>Fits in where OCP in a single node doesn't fit (e.g. HW restrictions)</div> <div>IoT / data collection gateways</div> <div>uCPE (customer premise equipment)</div>		<div>In-vehicle field operations</div> <div>Telco 5G sparsely populated areas</div> <div>In-field single server operations</div> <div>Disconnected environments</div>		<div>Telco 5G far edge - RAN</div> <div>IoT / data collection gateways</div>	<div>Telco 5G near edge &amp; MEC</div> <div>Edge AI &amp; Data pipelining</div> <div>Smart manufacturing</div> <div>Remote office</div> <div>Disconnected clusters</div>

\* Roadmap, not a product yet

1. RHEL for Edge introduction
2. Creating an RHEL for Edge image
3. Deploying the RHEL for Edge image
4. Upgrading the RHEL for Edge image
5. Serverless APP with Podman
6. Upgrading the APP automatically





# **1. RHEL for Edge introduction**

2. Creating an RHEL for Edge image

3. Deploying the RHEL for Edge image

4. Upgrading the RHEL for Edge image

5. Serverless APP with Podman

6. Upgrading the APP automatically



# What's the difference?

**Similar to containers:**  
System boots from read-only disk, user data is kept on a different volumes

## RED HAT® ENTERPRISE LINUX®

### General purpose Operating System

- Customizable root directory
- Packages are integrated with OS directory
- Rollbacks based on OS images or backups
- Traditional update system
- yum/dnf packet managers systems

## RED HAT® ENTERPRISE LINUX for Edge

### Immutable Operating System

- Immutable (read-only) directories
- Package are isolated from root directory
- Package installs create layers easy to rollback
- Efficient Over-the-air updates
- Support for multiple OS branches and repositories
- rpm-ostree hybrid tree/package system





# What's the benefit of RHEL for Edge



## Simplified management

**Secure and scale** with the benefits of zero-touch provisioning, fleet health visibility, and quick security remediations throughout the whole lifecycle



## Platform consistency

Easily create purpose-built OS images optimized for the architectural challenges inherent at edge. It makes the **system more reliable and predictable.**



## Efficient over-the-air updates

Updates **transfer significantly less data** and are ideal for remote sites with limited or intermittent connectivity



## Unattended resilience

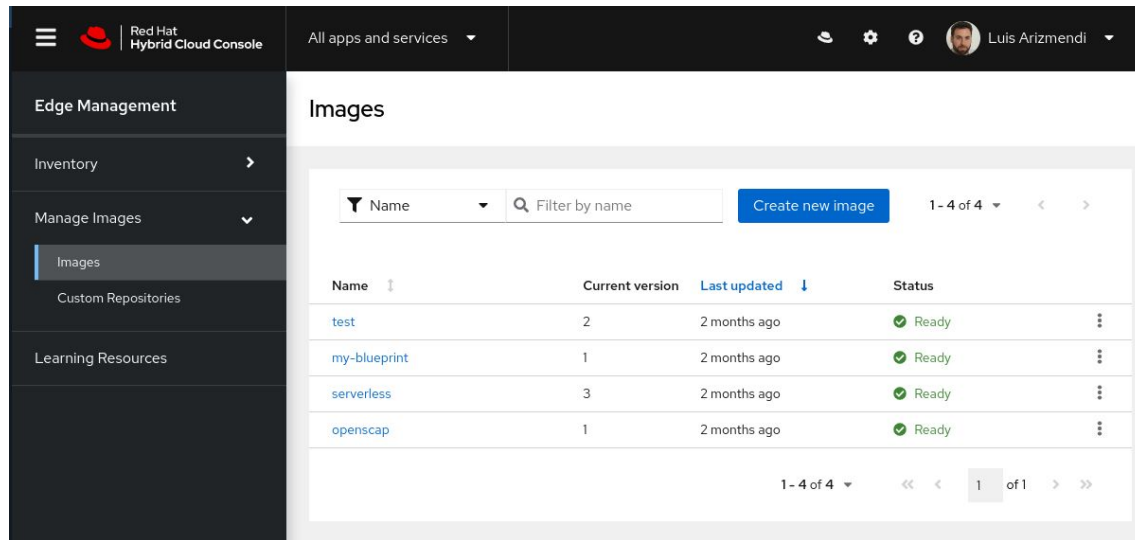
Application specific health checks detect conflicts and **automatically rollback** an OS update, preventing downtime

1. RHEL for Edge introduction
- 2. Creating an RHEL for Edge image**
3. Deploying the RHEL for Edge image
4. Upgrading the RHEL for Edge image
5. Serverless APP with Podman
6. Upgrading the APP automatically



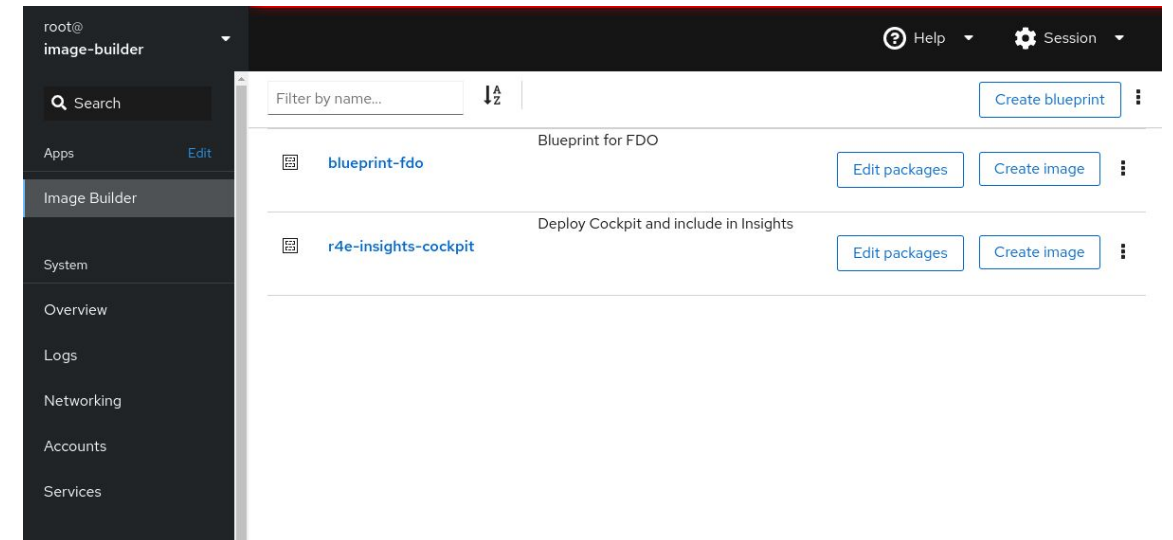
# FDO and RHEL for Edge images

Technology  
Preview



**console.redhat.com**

FDO client +  
(roadmap) image manufacturing process



**RHEL image builder**

FDO client +  
Image manufacturing process

# Image-based Operating System

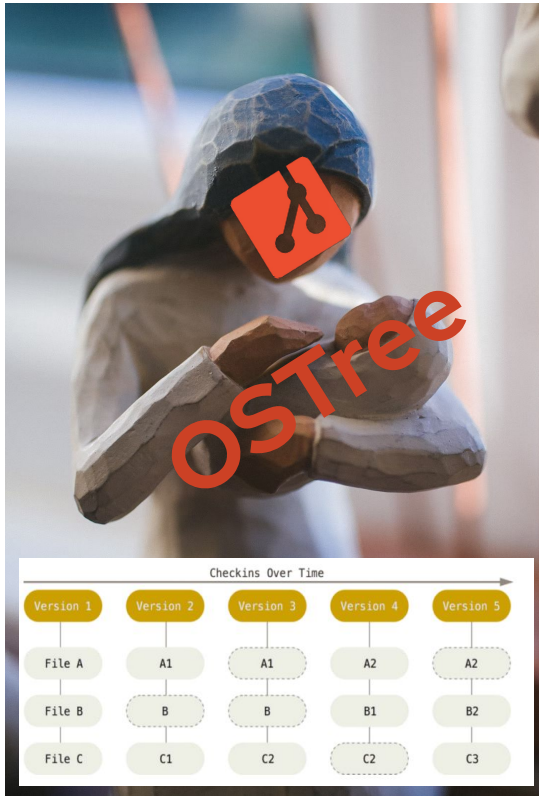
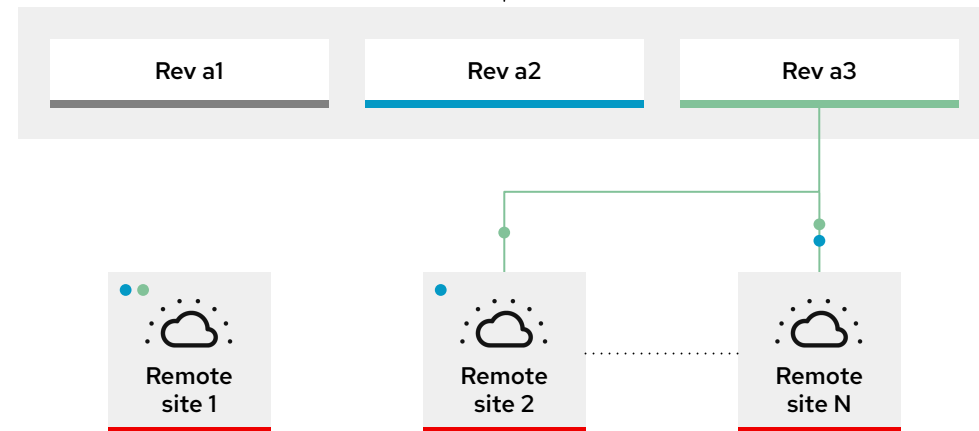


Image builder



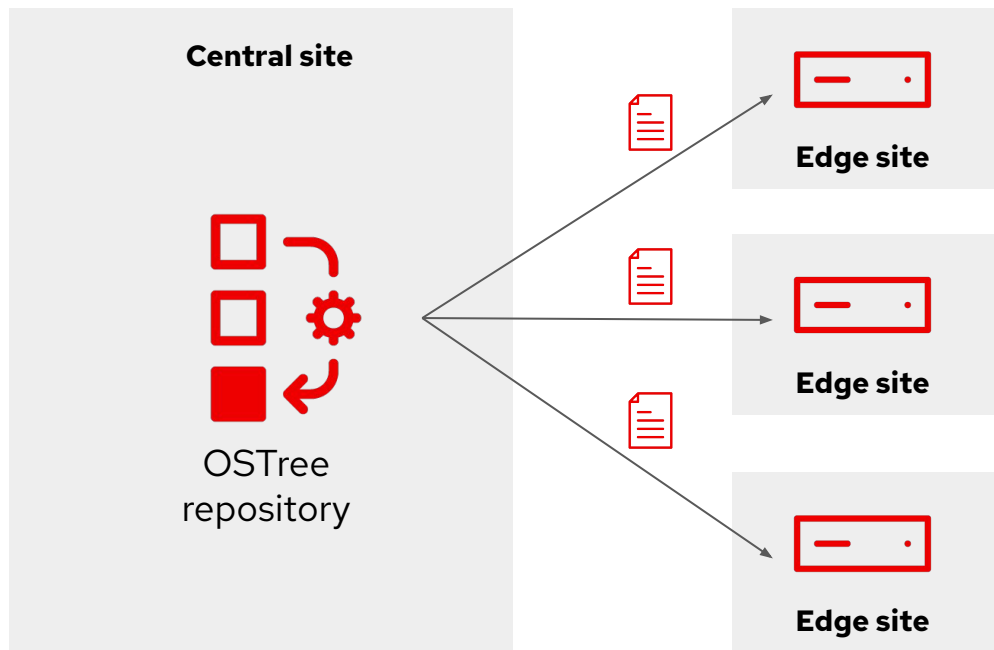
Update mirror



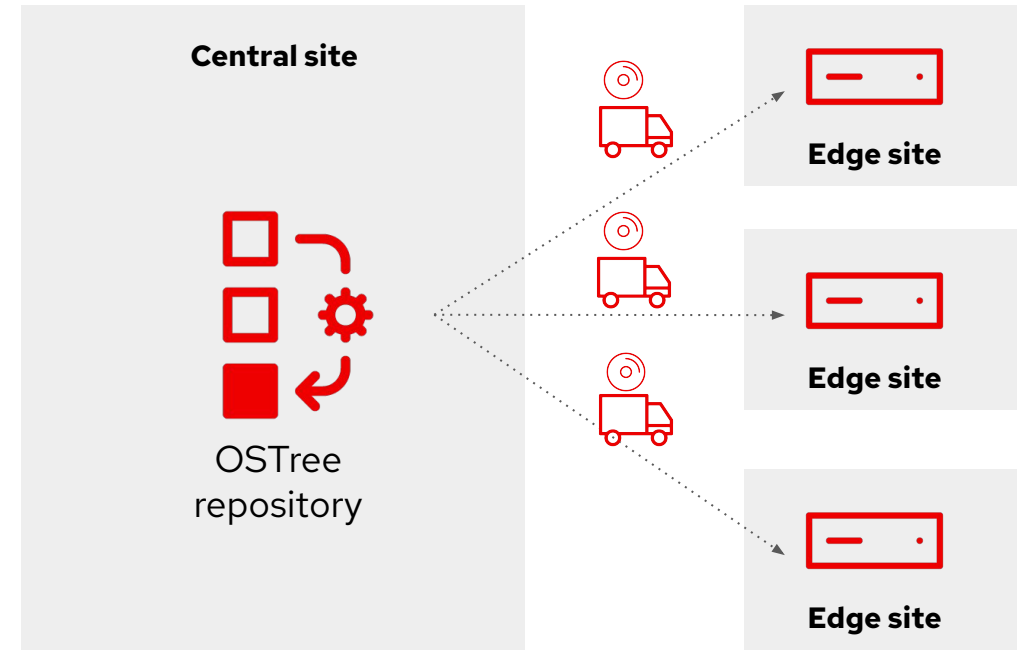
1. RHEL for Edge introduction
2. Creating an RHEL for Edge image
- 3. Deploying the RHEL for Edge image**
4. Upgrading the RHEL for Edge image
5. Serverless APP with Podman
6. Upgrading the APP automatically



# How do you deploy RHEL for Edge images?



**Network based deployment**

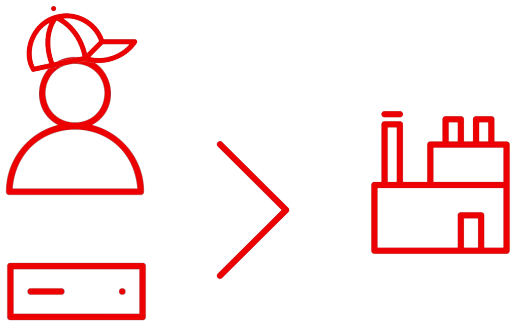


**Non-network based deployment**

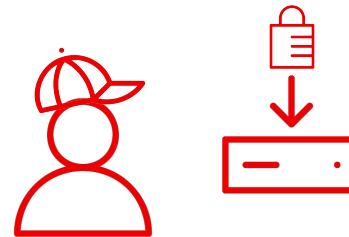




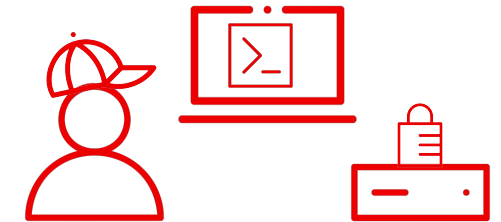
## Traditional onboarding



① Send device and **specialist** to edge location



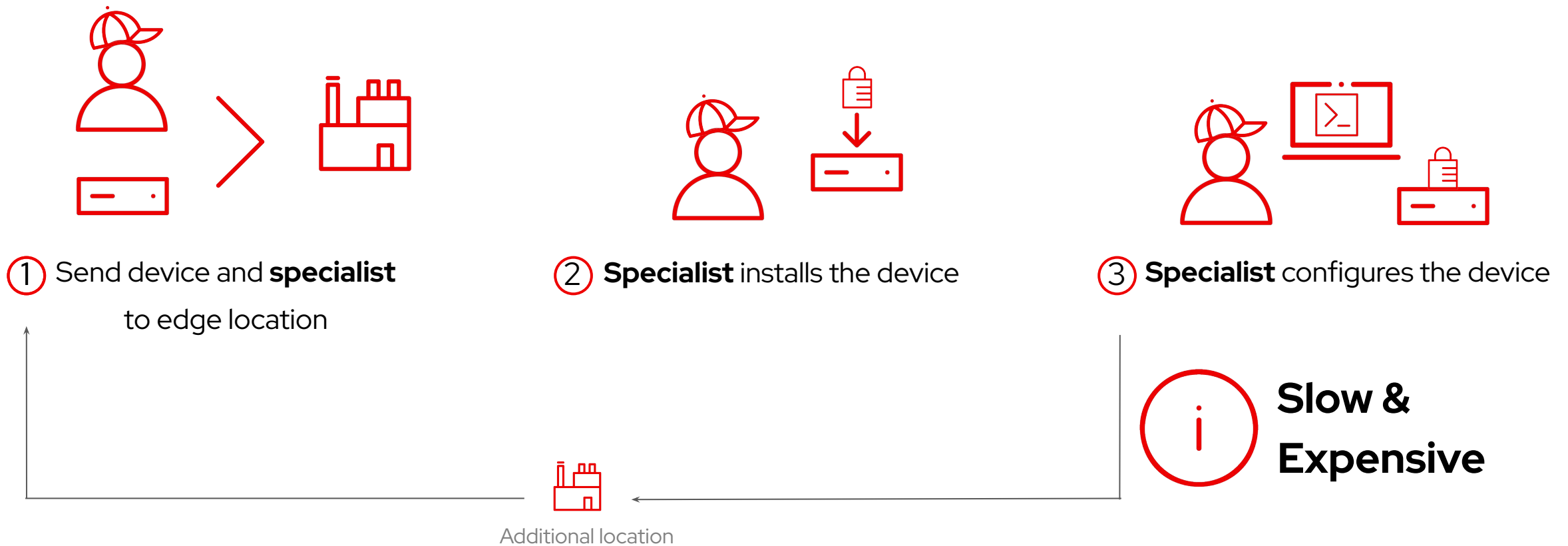
② **Specialist** installs the device



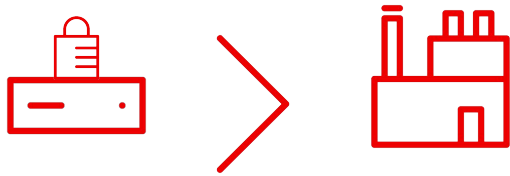
③ **Specialist** configures the device



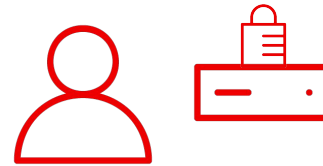
## Traditional onboarding



## "Hands-off" onboarding



① Send device to edge location



② **Anyone** connects the device

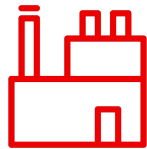


③ Device auto-installs and configures itself

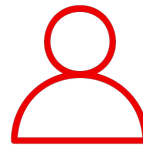
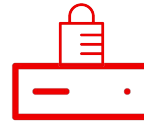


## "Hands-off" onboarding

**Risk!**



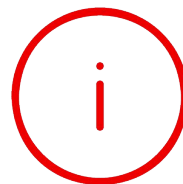
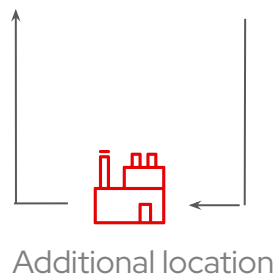
**Risk!**



① Send device to edge location

② **Anyone** connects the device

③ Device auto-installs and configures itself

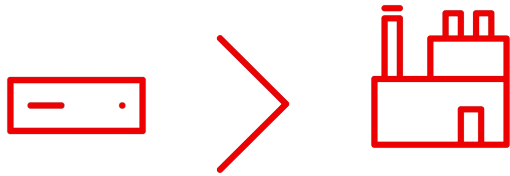


**Complex**

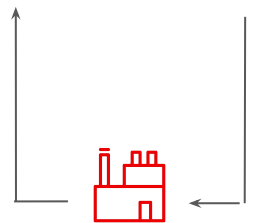
(Device "flavours", config changes, ...)



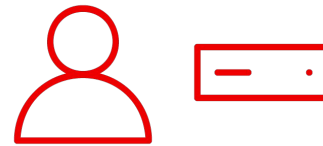
## "Late binding" onboarding



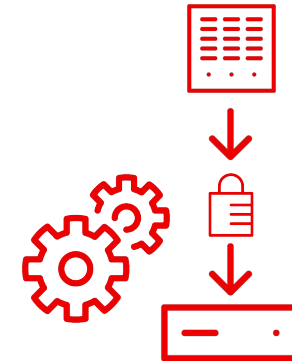
① Send device to edge location



Additional location



② **Anyone** connects the device



③ Device auto-installs and configures itself

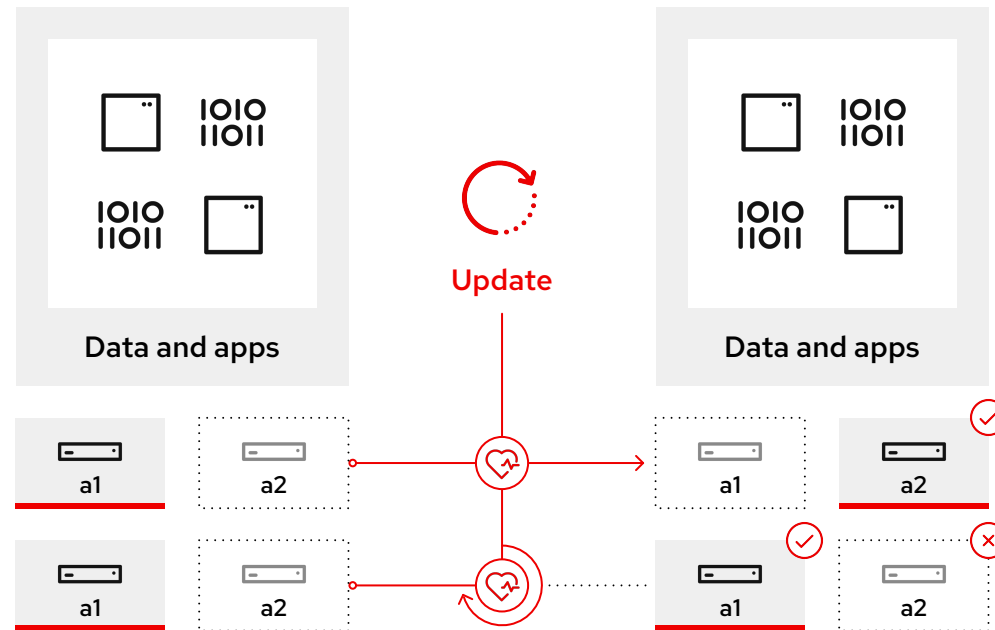


1. RHEL for Edge introduction
2. Creating an RHEL for Edge image
3. Deploying the RHEL for Edge image
- 4. Upgrading the RHEL for Edge image**
5. Serverless APP with Podman
6. Upgrading the APP automatically





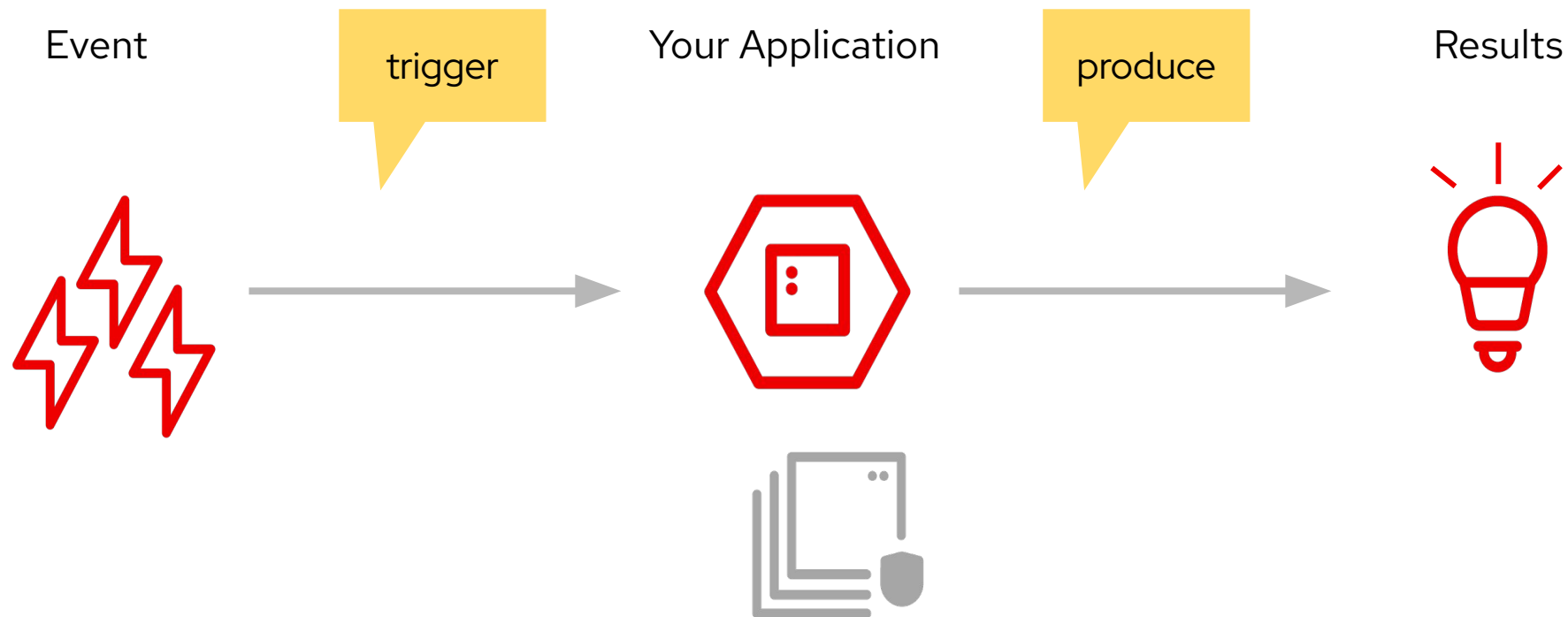
# How does atomic upgrades work?



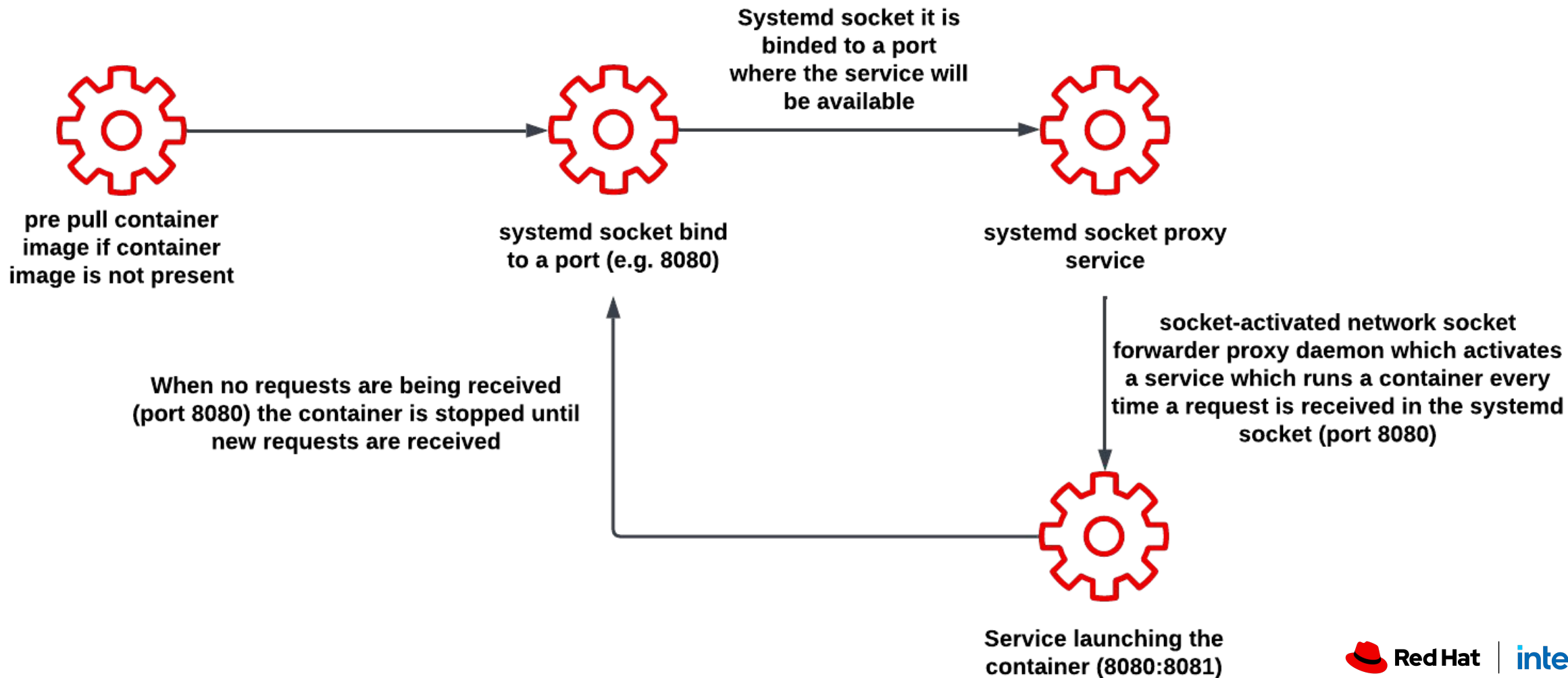
1. RHEL for Edge introduction
2. Creating an RHEL for Edge image
3. Deploying the RHEL for Edge image
4. Upgrading the RHEL for Edge image
- 5. Serverless APP with Podman**
6. Upgrading the APP automatically



## What's the idea behind using serverless at edge locations?



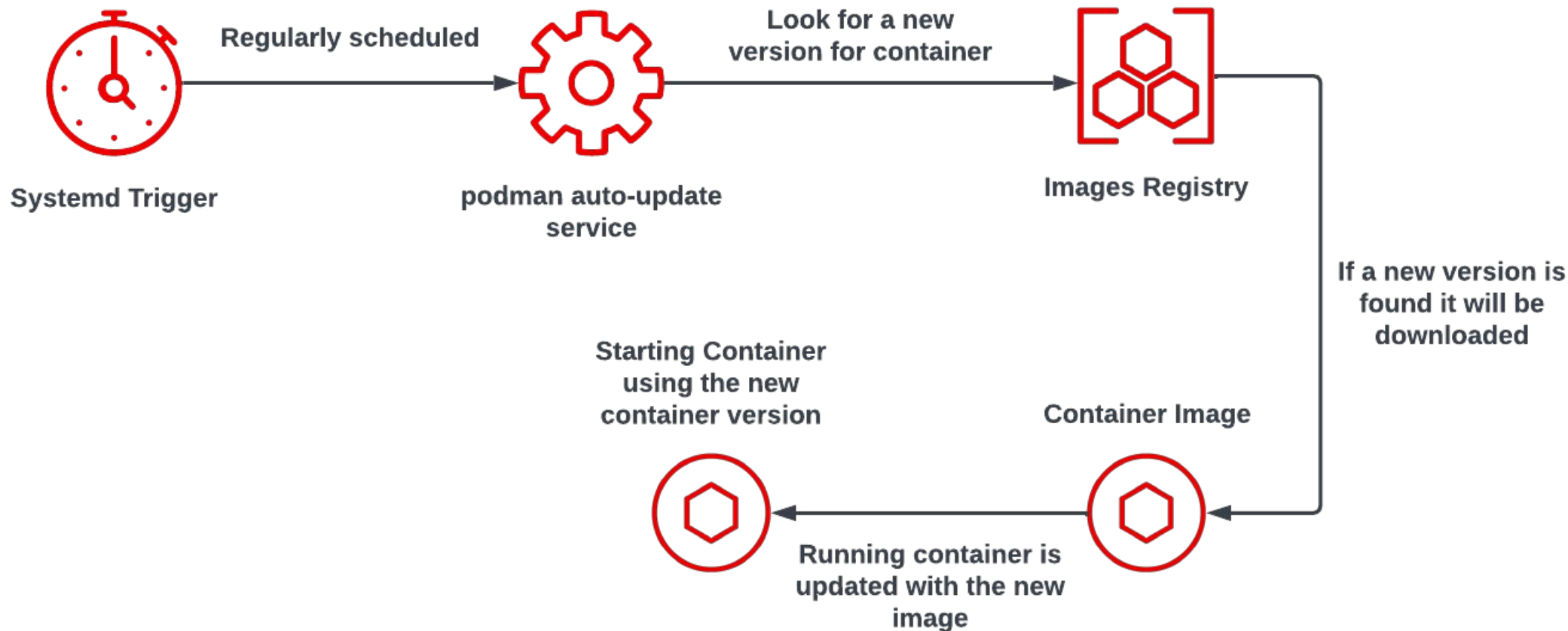
## How does a serverless application works?



1. RHEL for Edge introduction
2. Creating an RHEL for Edge image
3. Deploying the RHEL for Edge image
4. Upgrading the RHEL for Edge image
5. Serverless APP with Podman
- 6. Upgrading the APP automatically**

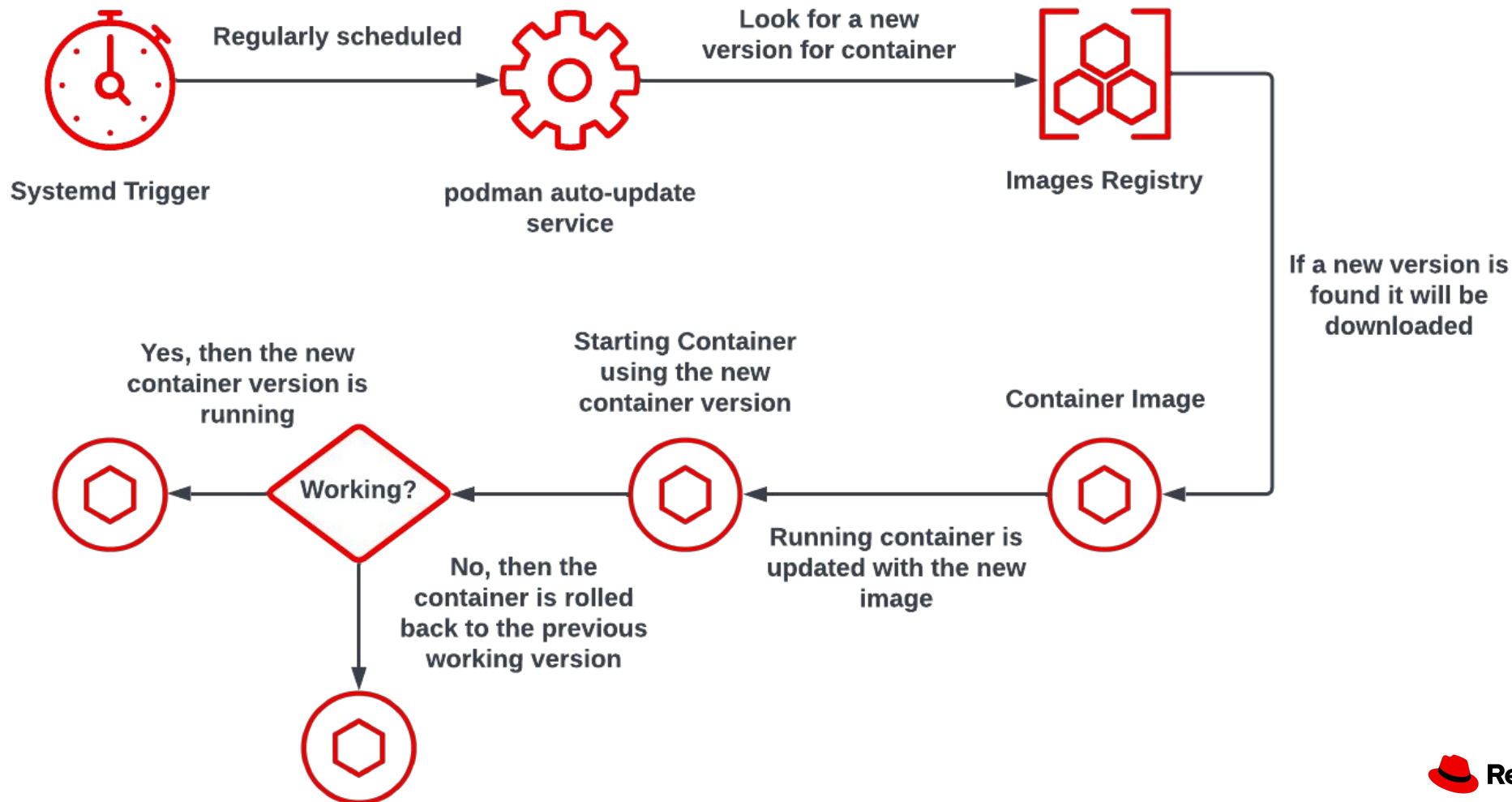


## How does automatic container upgrade work?





## How does automatic container upgrade work? (and rollback!!)



1. RHEL for Edge introduction
2. Creating an RHEL for Edge image
3. Deploying the RHEL for Edge image
4. Upgrading the RHEL for Edge image
5. Serverless APP with Podman
6. Upgrading the APP automatically



- [Edge device onboarding: What architects need to consider](#)
- [Edge device onboarding with the FDO specification](#)
- [How to onboard edge devices at scale with FDO and Linux](#)
- [Automating RHEL for Edge image rollback with GreenBoot](#)
- <https://jadebustos.github.io/workshops-rhel9>
- <https://github.com/luisarizmendi/rhel-edge-quickstart>
- <https://lab.redhat.com/>

Red Hat  
**Summit**

**Connect**

# Thank you



[linkedin.com/company/red-hat](https://linkedin.com/company/red-hat)



[facebook.com/redhatinc](https://facebook.com/redhatinc)



[youtube.com/user/RedHatVideos](https://youtube.com/user/RedHatVideos)



[twitter.com/RedHat](https://twitter.com/RedHat)