Red Hat Open Tour 2022

How to secure your business against cyber criminals

Tommi Sohlberg

Sr. Solution Architect

Agenda - Security trends

- **Trend 1: Digital Supply Chain Risk**

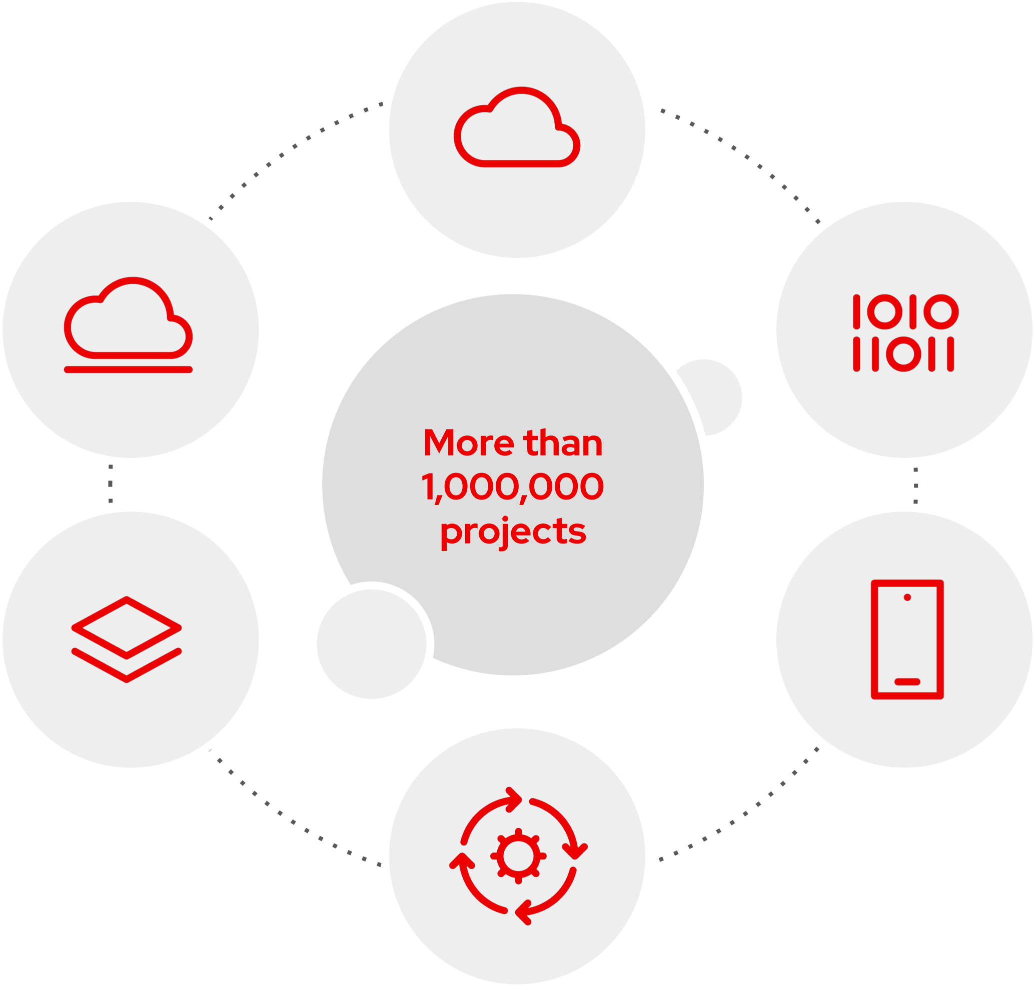
  Cybercriminals have discovered that attacks on the digital supply chain can provide a high return on investment. As vulnerabilities such as Log4j spread through the supply chain, more threats are expected to emerge. In fact, Gartner predicts that by 2025, 45% of organizations worldwide will have experienced attacks on their software supply chains, a three-fold increase from 2021.

- **Trend 2: Attack Surface Expansion**

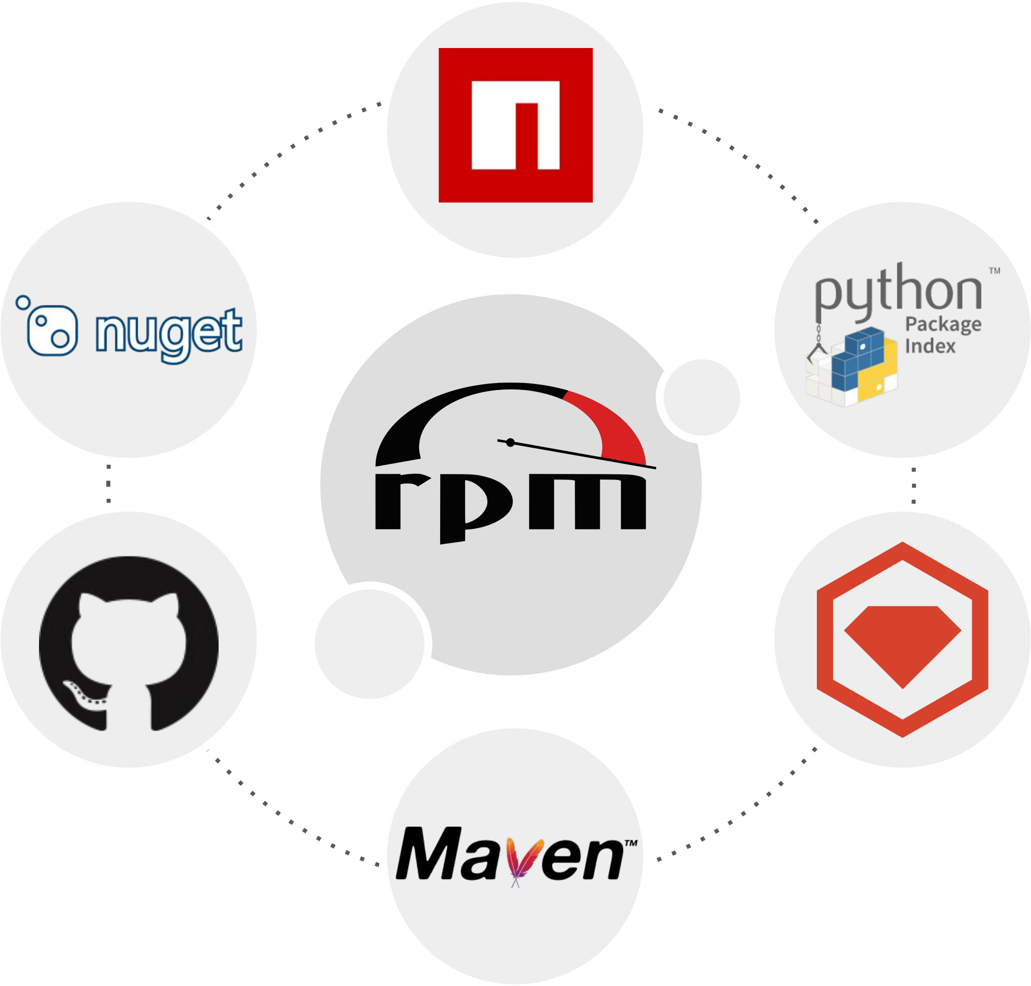
  Enterprise attack surfaces are expanding. Risks associated with the use of cyber-physical systems and IoT, open-source code, cloud applications, complex digital supply chains, social media and more have brought organizations' exposed surfaces outside of a set of controllable assets.

https://www.gartner.com/en/newsroom/press-releases/2022-03-07-gartner-identifies-top-security-and-risk-management-trends-for-2022

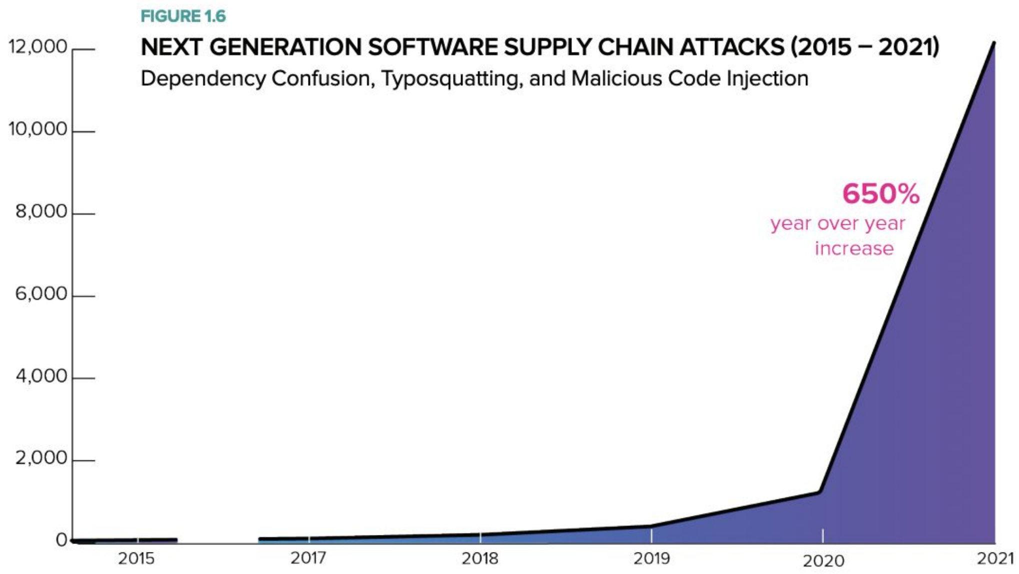
Open source fuels rapid innovation


More than 1,000,000 projects

Where open source lives



Attacks are increasing year on year & targeting OSS projects

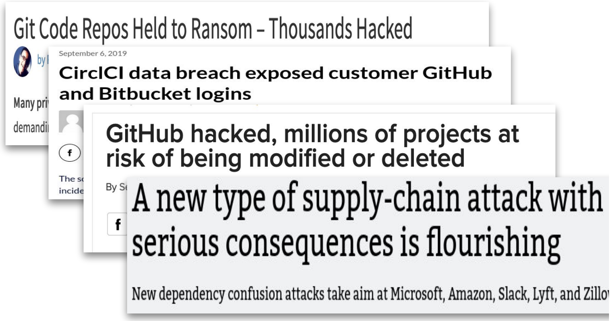


650%

Increase in supply chain attacks in 2021

[Sonatype's State of the Software Supply Chain]()

Software supply chains attacks



Git Code Repos Held to Ransom – Thousands Hacked

by

Many pri

demandi

September 6, 2019

CircICI data breach exposed customer GitHub and Bitbucket logins

The sc

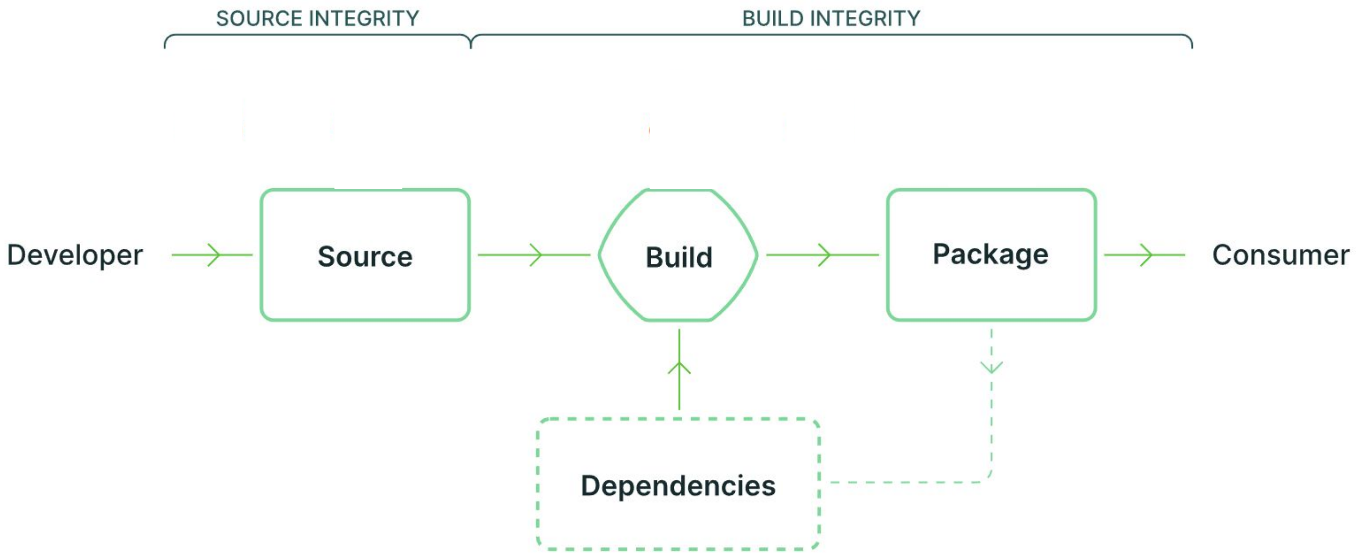
incide

GitHub hacked, millions of projects at risk of being modified or deleted

By Se

A new type of supply-chain attack with serious consequences is flourishing

New dependency confusion attacks take aim at Microsoft, Amazon, Slack, Lyft, and Zillo



- Replay / freeze attacks
- Compromised keys
- Account Compromise
- Swapped hashes
- Compromise of build systems
- Easy reconnaissance (open configuration)
- Typosquatting
- Maintainer account takeover

Software supply chains attacks

Git Code Repos Held to Ransom – Thousands Hacked

September 6, 2019

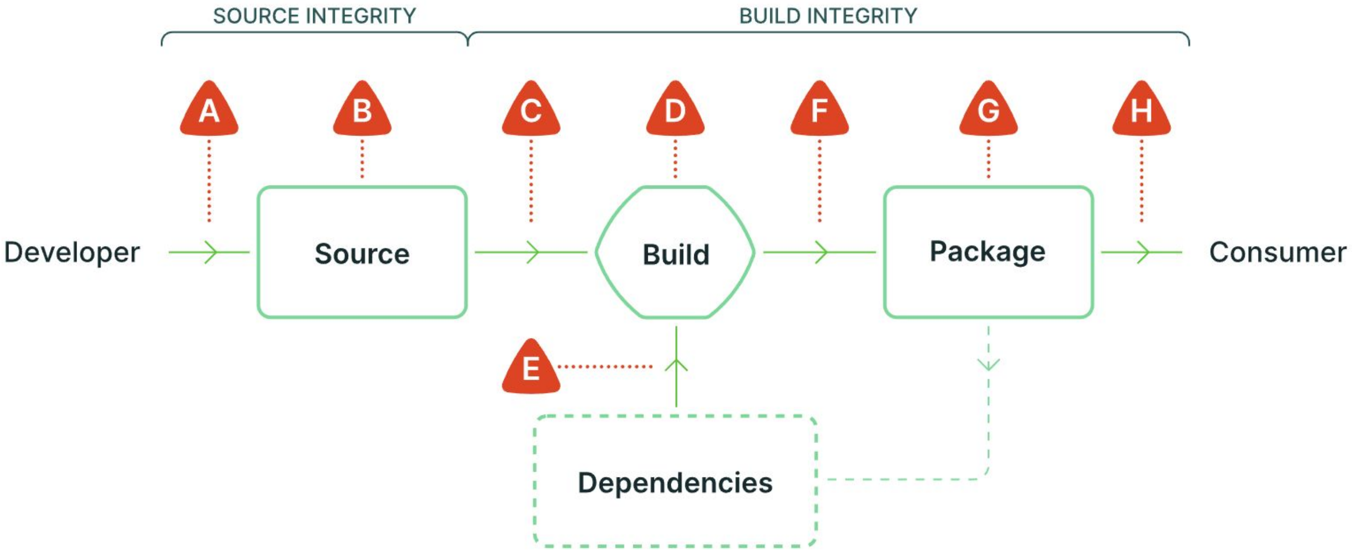
CirclCI data breach exposed customer GitHub and Bitbucket logins

GitHub hacked, millions of projects at risk of being modified or deleted

A new type of supply-chain attack with serious consequences is flourishing

New dependency confusion attacks take aim at Microsoft, Amazon, Slack, Lyft, and Zillo

- Replay / freeze attacks
- Compromised keys
- Account Compromise
- Swapped hashes
- Compromise of build systems
- Easy reconnaissance (open configuration)
- Typosquatting
- Maintainer account takeover



A Submit unauthorized change
B Compromise source repo

C Build from modified source
D Compromise build process
E Use compromised dependency

F Upload modified package
G Compromise package repo
H Use compromised package

So what should we do about it?

Supply Chain Control

The story of the supply chain is the story of how a vendor creates their offerings and from where they source their materials. **Your supply chain is not only what you make and how you make it**, but what things exist within the ecosystem of the system that provides that engine.

Undermanaged software can have costly impacts

6 million new versions

of OSS introduced in the past year; 37 million component versions now available[1]

650% increase

in open source software supply chain attacks[1]

$25 million

the predicted cost of a recent supply chain attack[2]

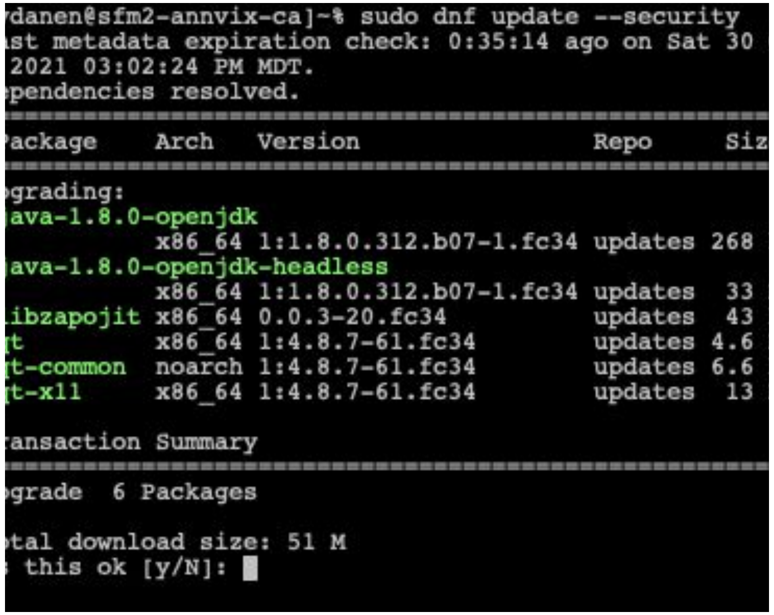
$2 billion

the cost of a data breach that resulted from an unpatched bug[3]

Source:
[1] Sonatype 2021 State of the software supply chain
[2] SolarWinds Expects Cyber Incident Costs Up To $25 Million In 2021
[3] Equifax to Pay at Least $650 Million in Largest-Ever Data Breach Settlement

Security considerations for open source software

- How are new vulnerabilities in open source software discovered?
- What level of awareness exists around open source software in use?
- How are the security impact to the software you have assessed?
- How are fixes to the software in use addressed?
- Is the appropriate expertise to assess and remediate security issues in open source software available in-house?
- What about critical and immediate support?



```
danen@sfm2-annvix-ca]-% sudo dnf update --security
st metadata expiration check: 0:35:14 ago on Sat 30
 2021 03:02:24 PM MDT.
pendencies resolved.
=====================================================
ackage    Arch   Version                 Repo    Siz
=====================================================
grading:
ava-1.8.0-openjdk
          x86_64 1:1.8.0.312.b07-1.fc34 updates 268
ava-1.8.0-openjdk-headless
          x86_64 1:1.8.0.312.b07-1.fc34 updates  33
ibzapojit x86_64 0.0.3-20.fc34          updates  43
t         x86_64 1:4.8.7-61.fc34        updates 4.6
t-common  noarch 1:4.8.7-61.fc34        updates 6.6
t-x11     x86_64 1:4.8.7-61.fc34        updates  13

ansaction Summary
=====================================================
grade  6 Packages

tal download size: 51 M
 this ok [y/N]: █
```

> "The time to repurpose vulnerabilities into working exploits will be measured in hours and there's nothing you can do about it… except patch."

Fred House
Senior Director at FireEye, Inc.
(*McAfee Enterprise and FireEye 2022 Threat Predictions*)

Source:
https://www.fireeye.com/blog/executive-perspective/2021/10/mcafee-enterprise-fireeye-2022-threat-predictions.html

Backport or rebase?

For enterprise customers sensitive to change, backporting is the best choice

Backporting is taking an upstream change from a later version and applying it to an earlier version. Why backport?

- Isolate code changes to fix a specific issue
- Maintain API/ABI compatibility - existing apps continue to work without change
- Reduce risk of new vulnerabilities introduced in later versions

Rebasing is updating the version of software to the latest available upstream. Why rebase?

- Fixes are too complex to backport successfully
- Desirable functionality present in newer version
- Lack of expertise to backport successfully

Not vulnerable due to backporting

Security value of backports from Red Hat versus grabbing from upstream

CVE-2020-1967

Important OpenSSL

Vulnerability was introduced in OpenSSL version 1.1.1d, which we did not ship

CVE-2021-3345

Critical libgcrypt

Vulnerability was introduced in libgcrypt version 1.9.0, which we did not ship

CVE-2021-20226

Important kernel

Vulnerable upstream code was not introduced in any version we shipped

CVE-2020-8835

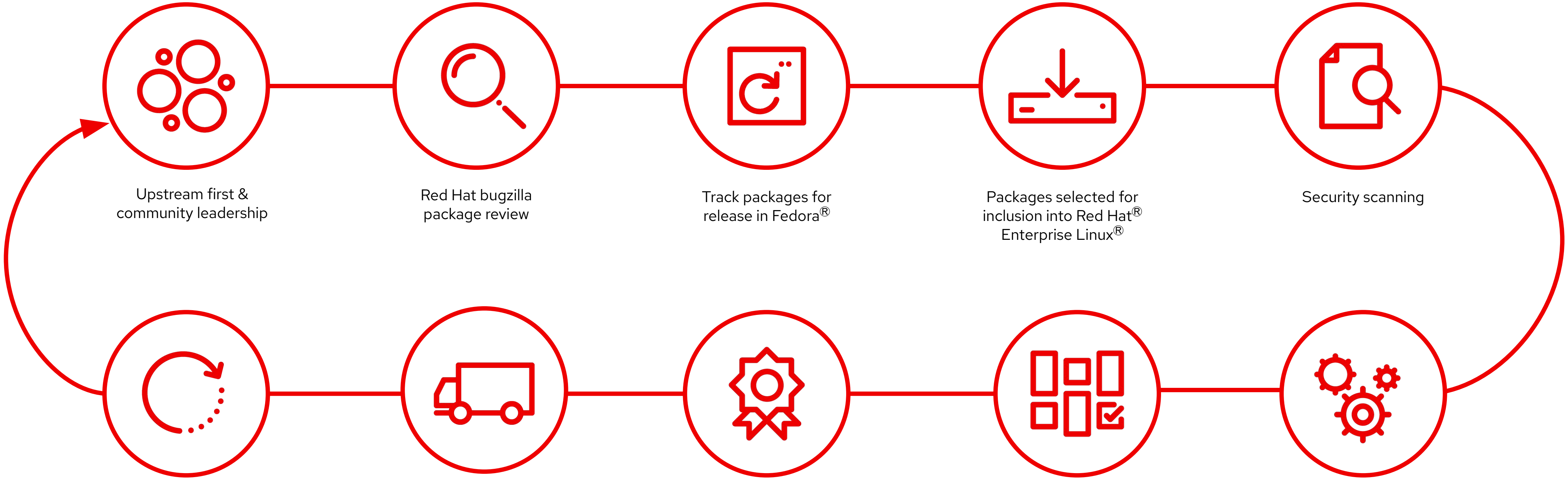
Important kernel

Vulnerable upstream code was not introduced in any version we shipped

Small data sample; there are many more examples where we did not take an upstream version or upstream patches due to our backporting policy.

Red Hat's software supply chain security

Reducing risk and making open source consumable for the enterprise



Upstream first & community leadership

Red Hat bugzilla package review

Track packages for release in Fedora®

Packages selected for inclusion into Red Hat® Enterprise Linux®

Security scanning

Compiler flags set for hardening and security

Extensive QE testing per release

All packages are digitally signed

Secure distribution

Continuous security updates

Mitigating supply chain security risk

Signing software helps, but it’s (still) hard

What if signing and key management were greatly simplified…

…and with open transparency

sigstore

In collaboration with

OpenSSF CHAINGUARD CISCO Google Hewlett Packard Enterprise PURDUE UNIVERSITY vmware

Sigstore - the Vision

Attestation of Software Supply Chain, from upstream commit to production runtime

At each step, everything is

- Cryptographically signed
- Leveraging a shared root of trust
- Backed by an append-only log

How can you use it?

Sign

Easy authentication and smart cryptography work in the background. Just push your code, sigstore can handle the rest.

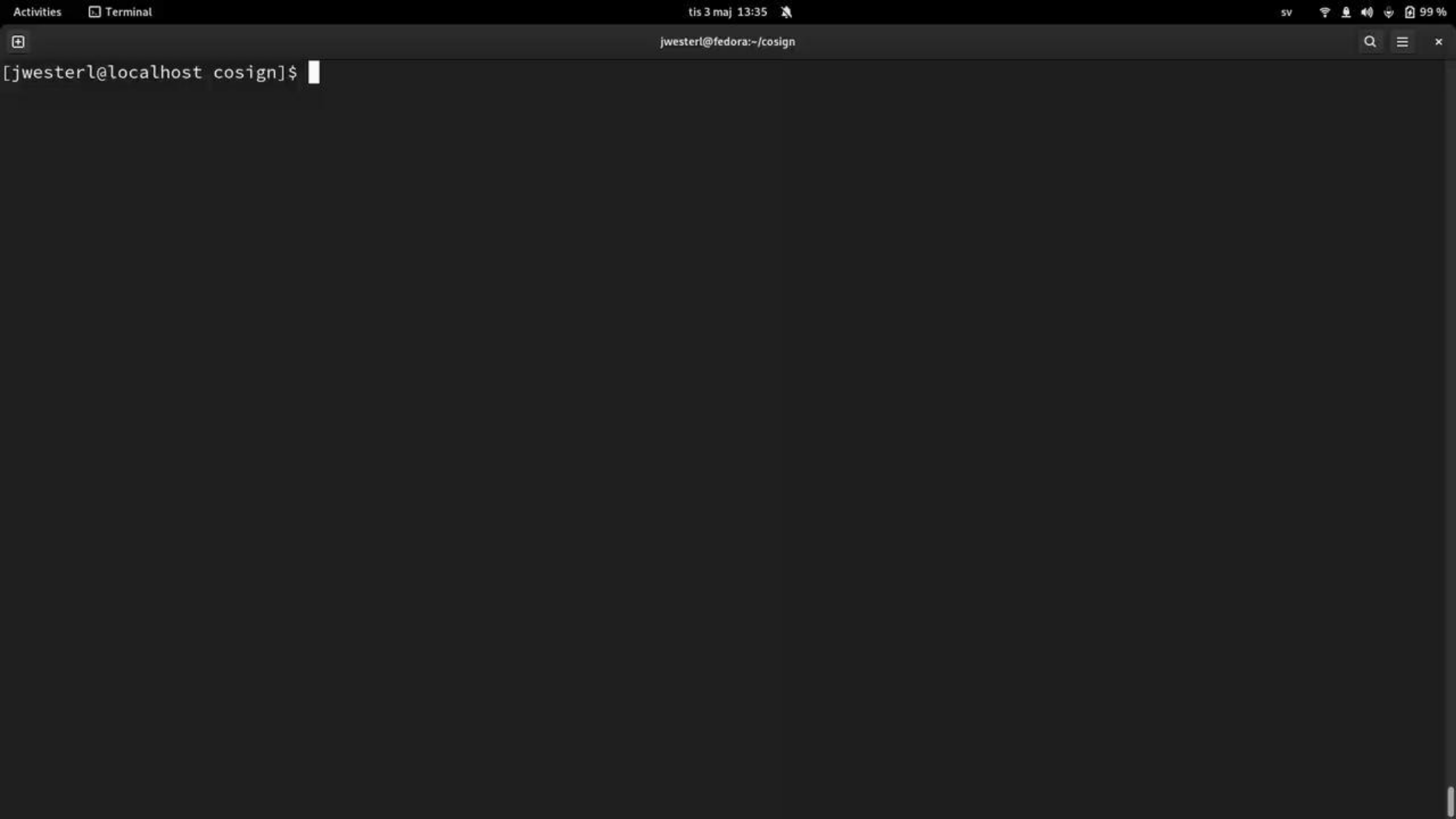
Verify

Rekor transparency logs store unique identification like who created it and where it was built, so you know it hasn’t been changed.

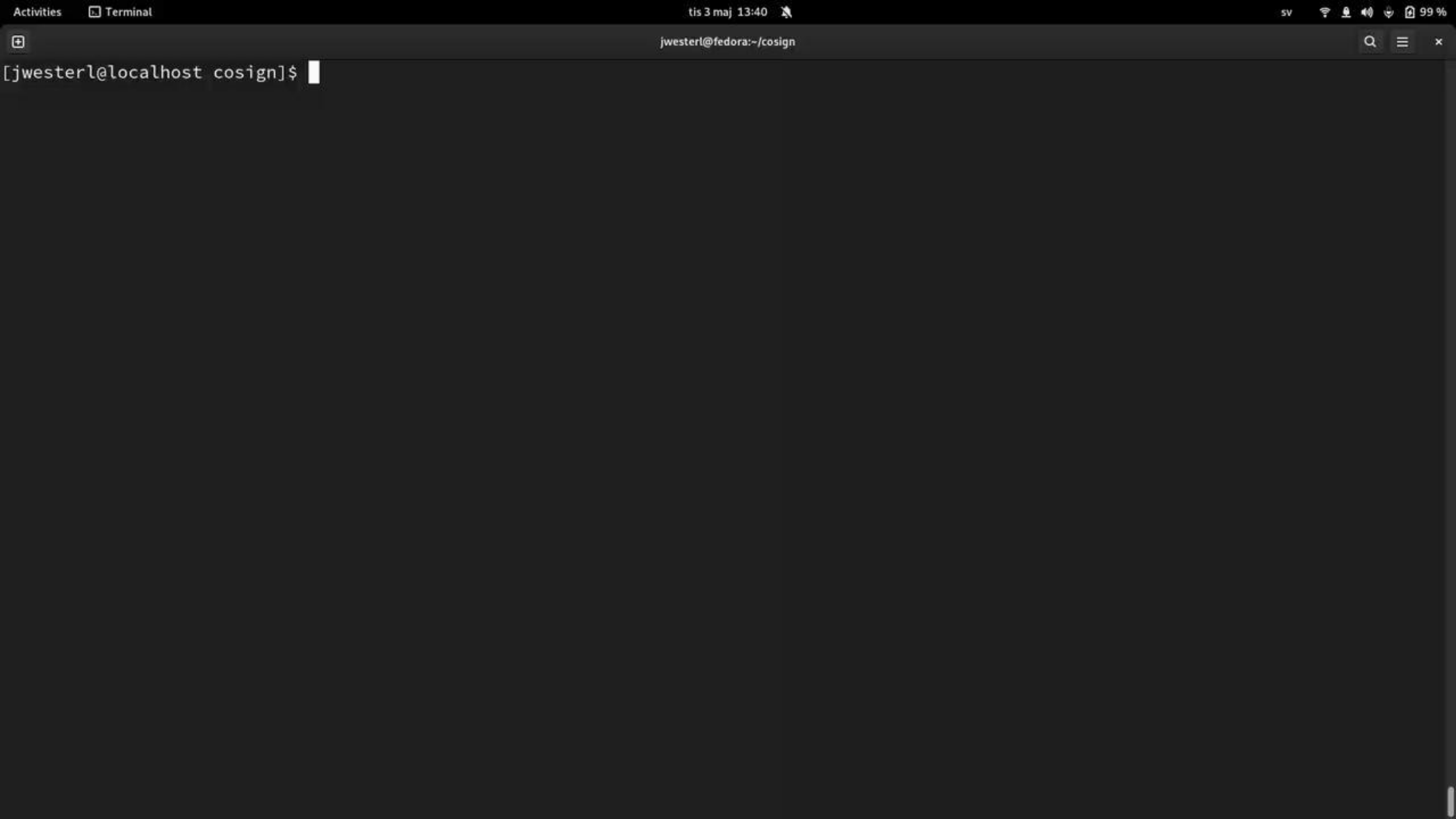
Monitor

Data stored in the logs is readily auditable, a foundation for future monitors and integrations to build into your security workflow

DEMO: Securing supply chain with sigstore



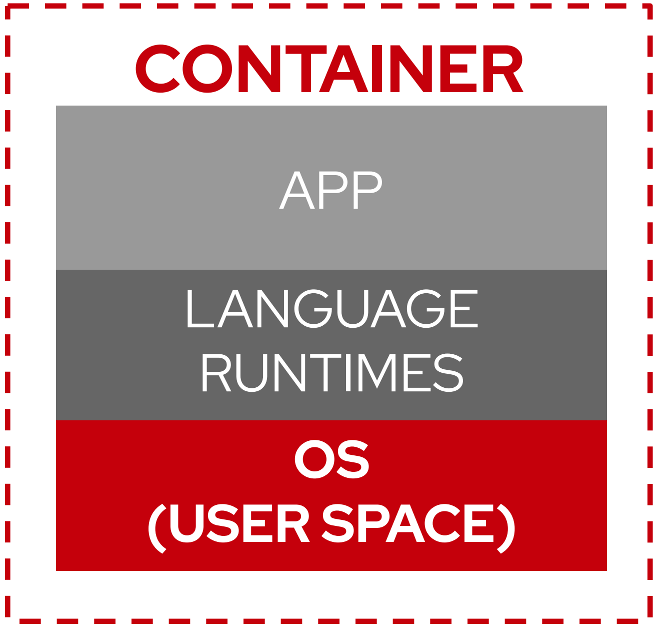
```
[jwesterl@localhost cosign]$ 
```



```
[jwesterl@localhost cosign]$
```

Signing is nice, but what should I sign?

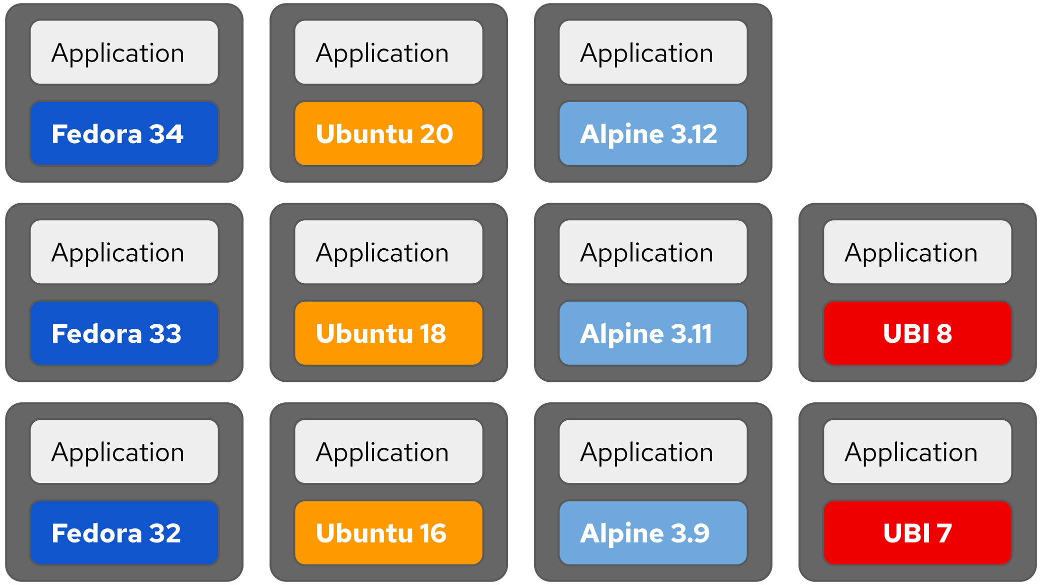
Red Hat Universal Base Image (UBI)



Trusted:

- Libraries
- Packaging format
- Core Utilities
- Security Response
- Patching
- Performance Response
- Technical Support
- More

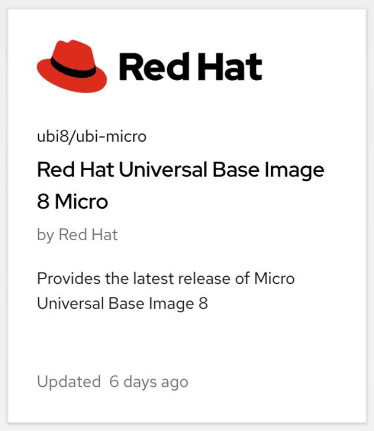
Wild Wild West



Application Fedora 34	Application Ubuntu 20	Application Alpine 3.12	
Application Fedora 33	Application Ubuntu 18	Application Alpine 3.11	Application UBI 8
Application Fedora 32	Application Ubuntu 16	Application Alpine 3.9	Application UBI 7

- 8 different versions of glibc
- 3 different versions of muslc
- 11 different versions of OpenSSL

Red Hat Universal Base Image (UBI)



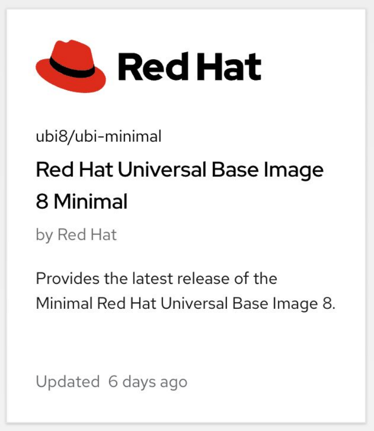
ubi8/ubi-micro

Red Hat Universal Base Image 8 Micro

by Red Hat

Provides the latest release of Micro Universal Base Image 8

Updated 6 days ago



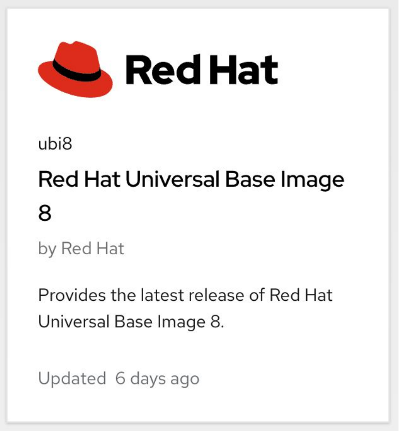
ubi8/ubi-minimal

Red Hat Universal Base Image 8 Minimal

by Red Hat

Provides the latest release of the Minimal Red Hat Universal Base Image 8.

Updated 6 days ago



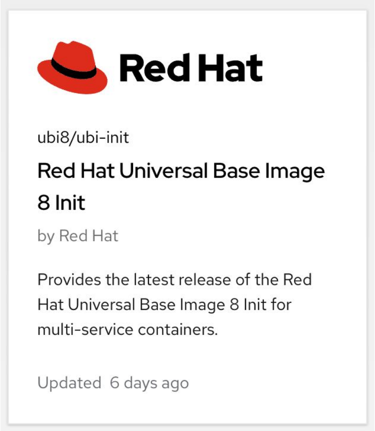
ubi8

Red Hat Universal Base Image 8

by Red Hat

Provides the latest release of Red Hat Universal Base Image 8.

Updated 6 days ago



ubi8/ubi-init

Red Hat Universal Base Image 8 Init

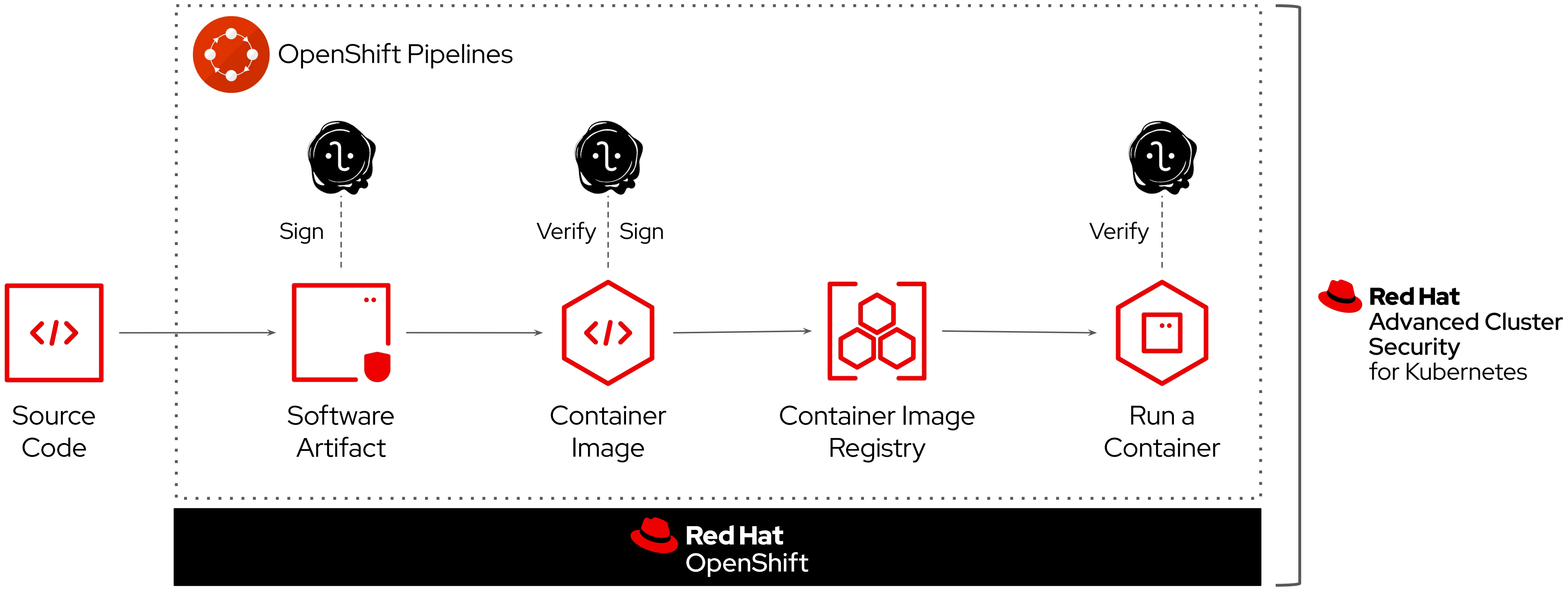
by Red Hat

Provides the latest release of the Red Hat Universal Base Image 8 Init for multi-service containers.

Updated 6 days ago

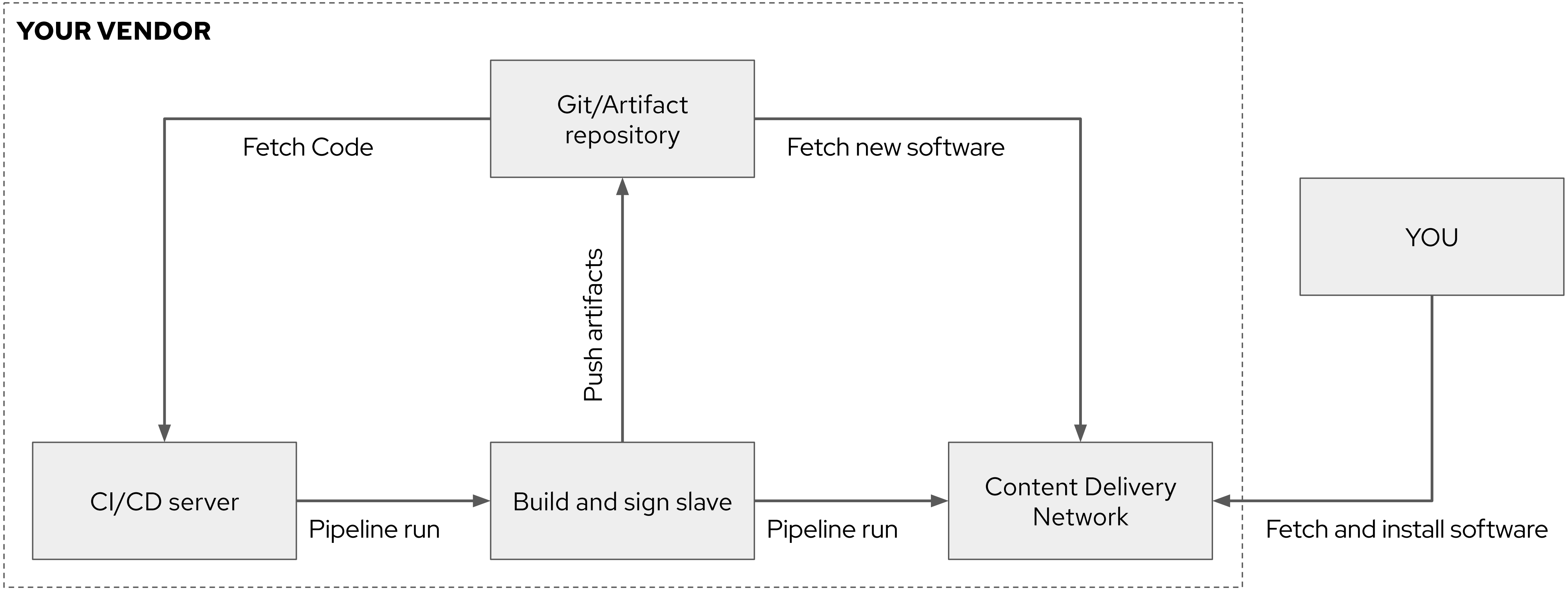
Choose image based on your requirements

Building a trust

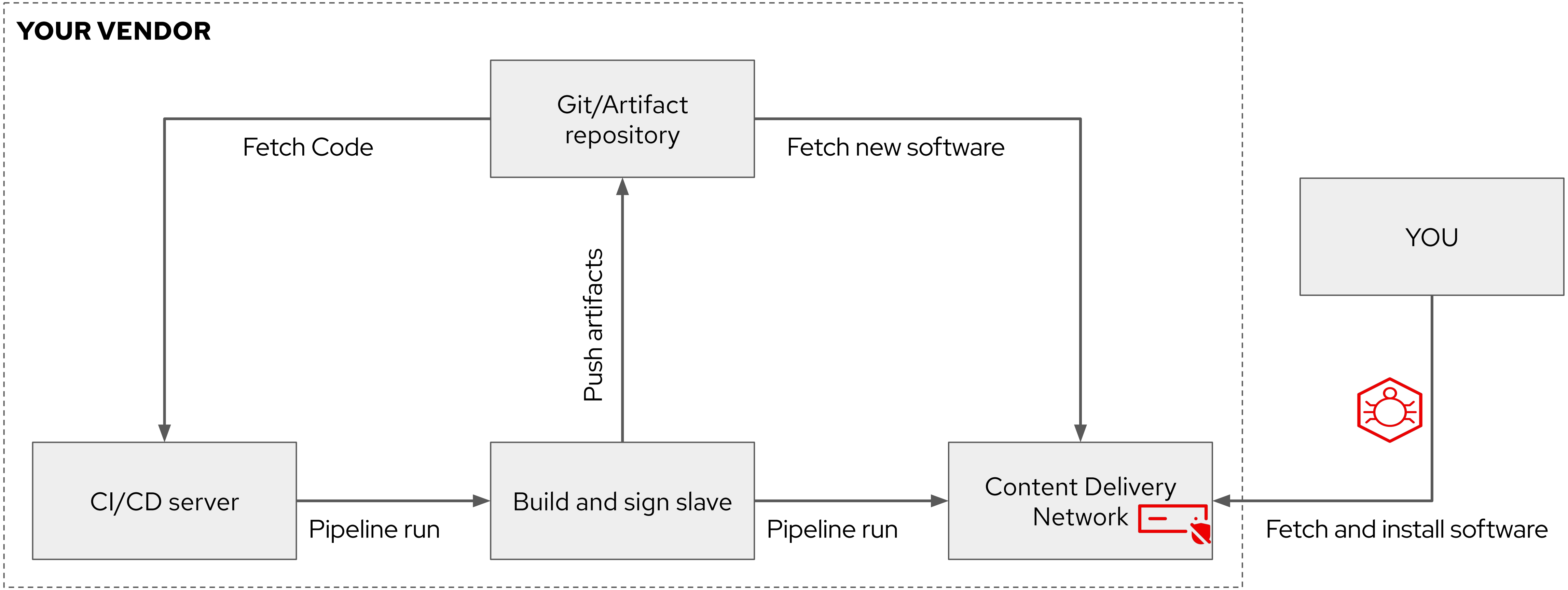


DEMO: Defending against supply chain attacks

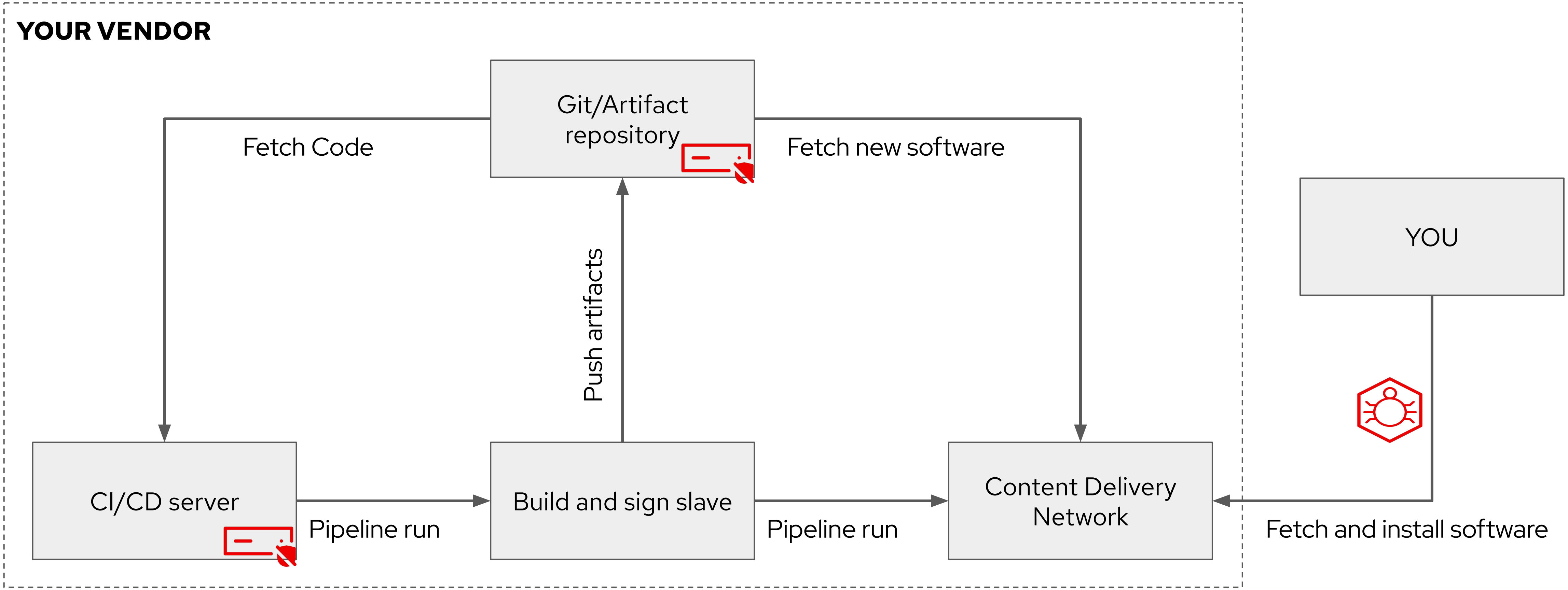
A software supply chain

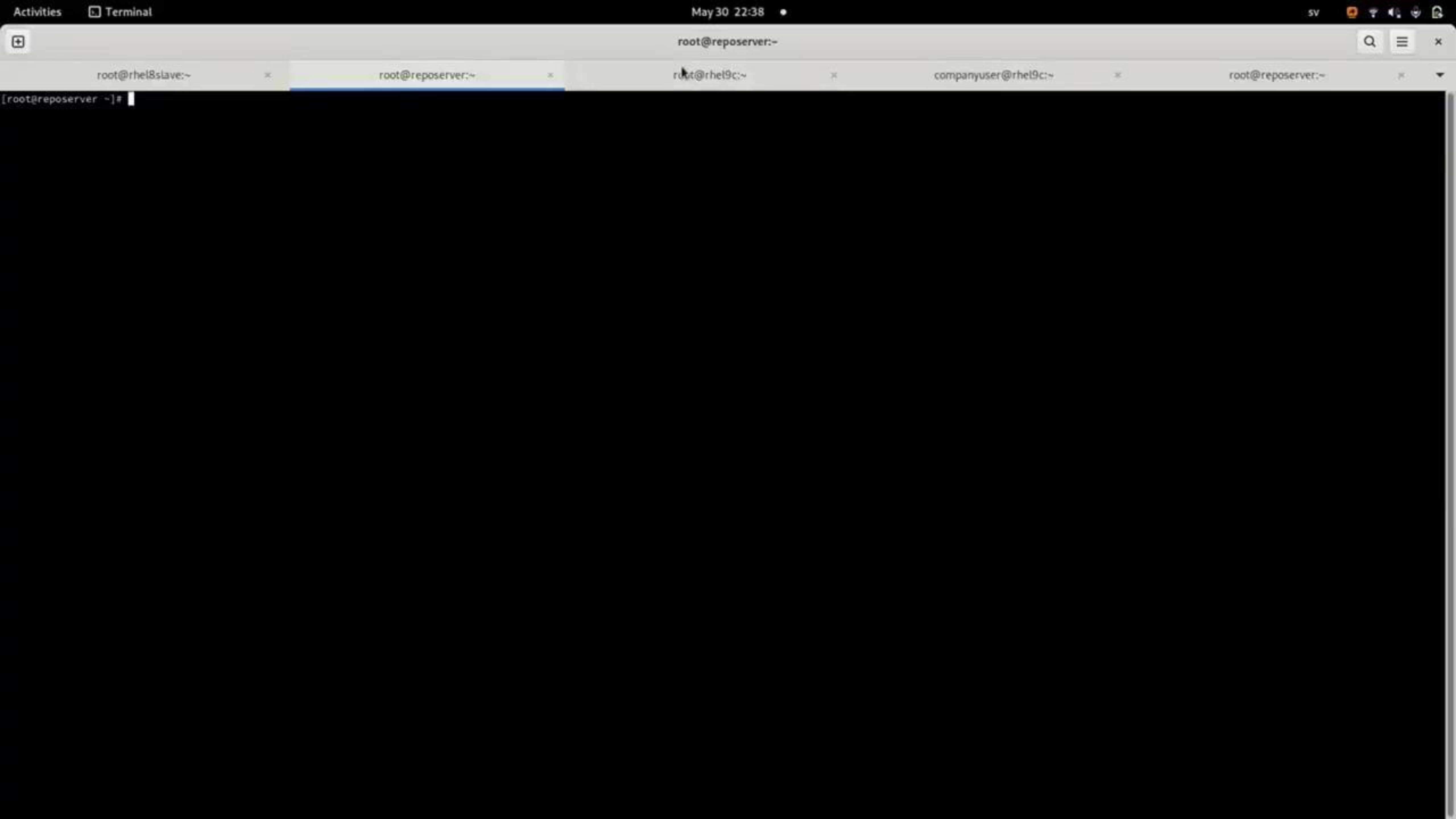


Attack 1: Content Delivery Network breached



Attack 2: Development process breached

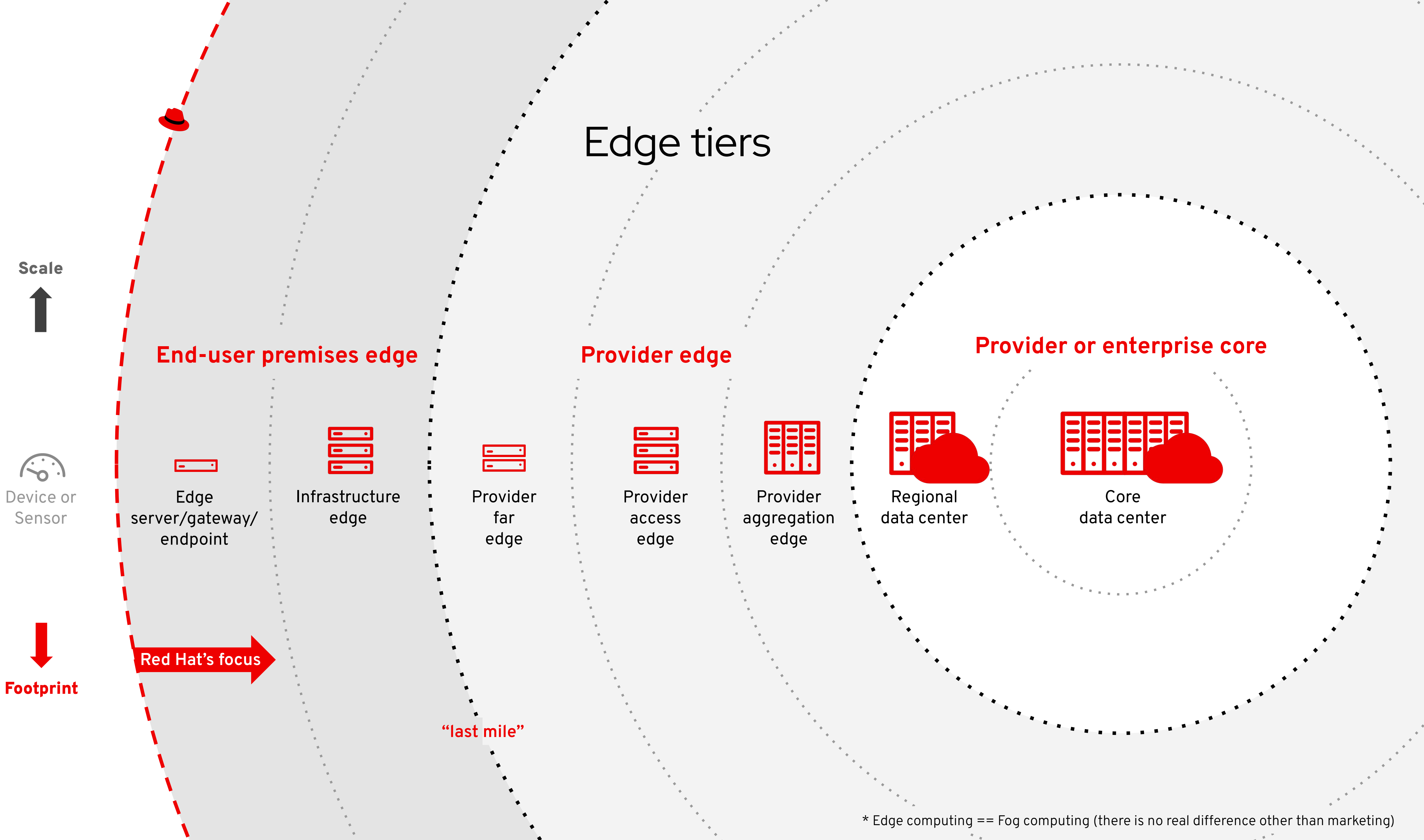




```
[root@reposerver ~]# 
```

Trend 2 – IoT/Edge

Edge tiers



Scale

Device or Sensor

Footprint

End-user premises edge

Edge server/gateway/endpoint

Infrastructure edge

Red Hat’s focus

“last mile”

Provider edge

Provider far edge

Provider access edge

Provider aggregation edge

Provider or enterprise core

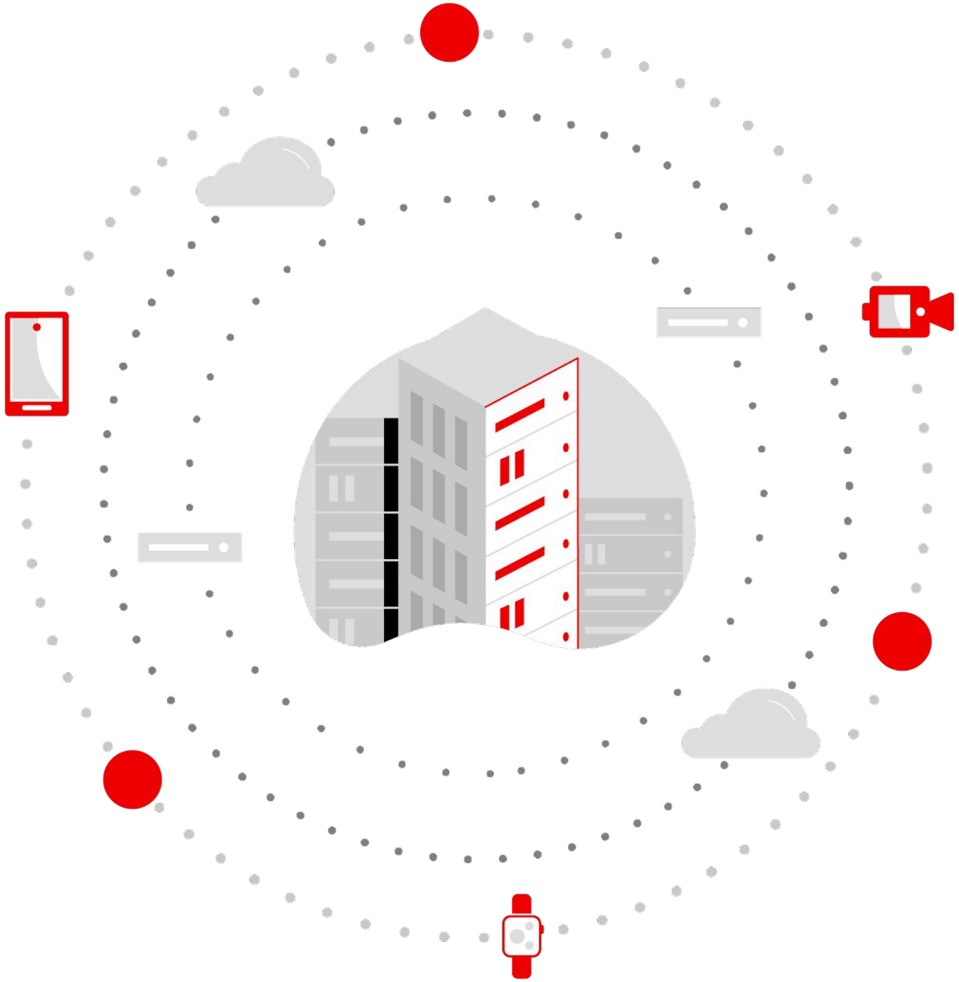
Regional data center

Core data center

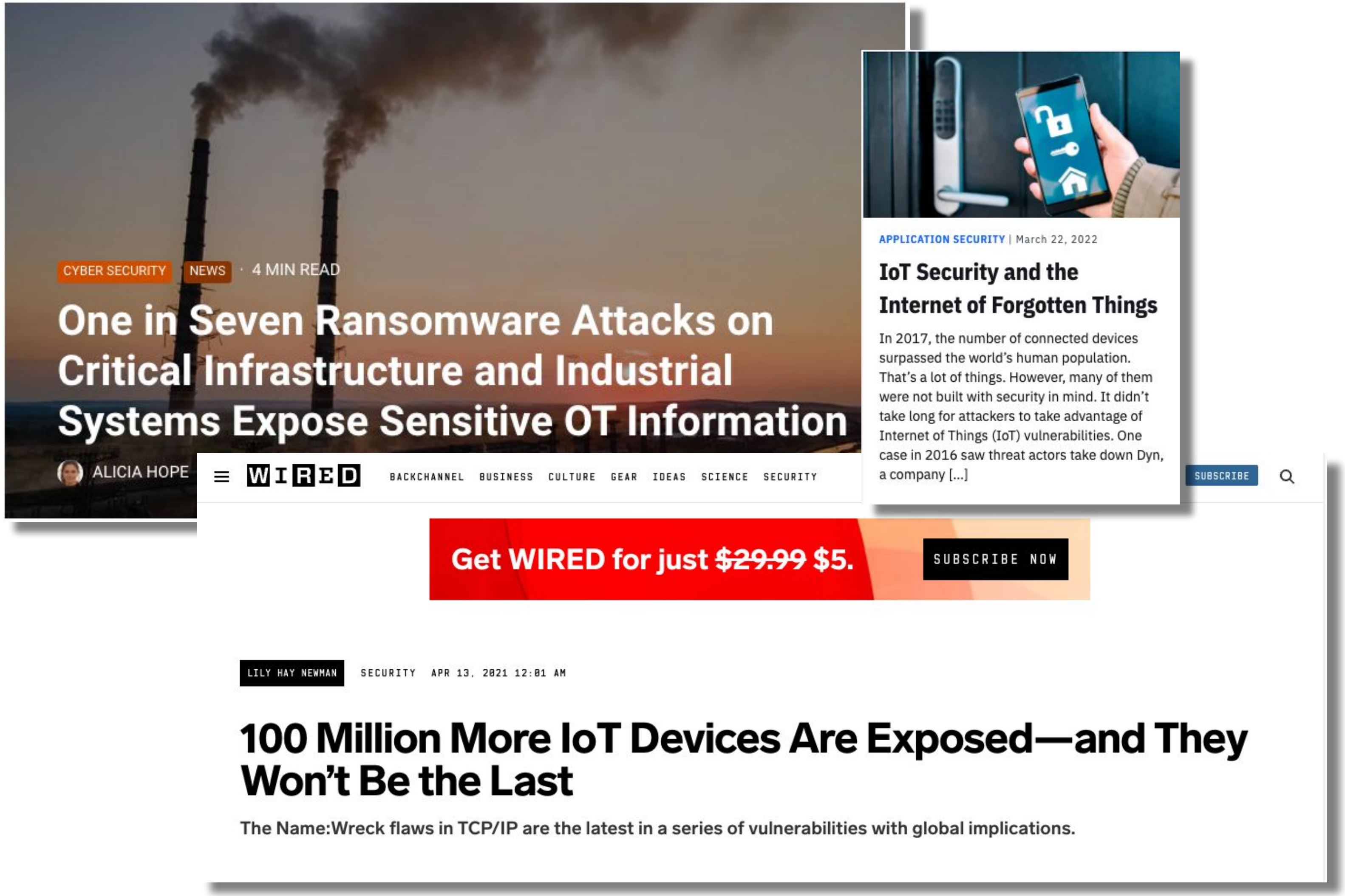
* Edge computing == Fog computing (there is no real difference other than marketing)

"800% increase in the number of apps deployed at the edge."[2]

"By 2025 more than 50% of enterprise-managed data will be created and processed outside the data center or cloud."[3]



[1] IDC: IDC FutureScape: Worldwide IT Industry 2020 Predictions, Doc # US45599219, October 2019
[2] Gartner: Predicts 2022: The Distributed Enterprise Drives Computing to the Edge October 2021



One in Seven Ransomware Attacks on Critical Infrastructure and Industrial Systems Expose Sensitive OT Information

ALICIA HOPE

| March 22, 2022

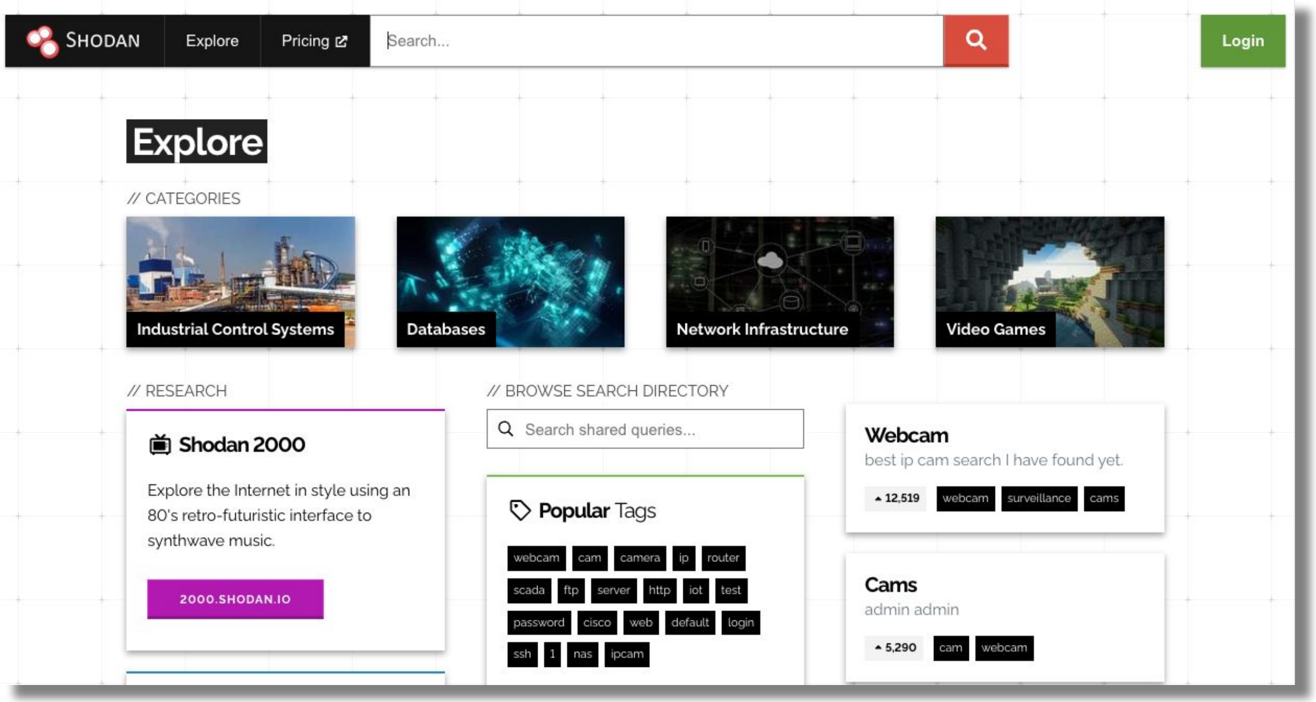
IoT Security and the Internet of Forgotten Things

In 2017, the number of connected devices surpassed the world's human population. That's a lot of things. However, many of them were not built with security in mind. It didn't take long for attackers to take advantage of Internet of Things (IoT) vulnerabilities. One case in 2016 saw threat actors take down Dyn, a company [...]

LILY HAY NEWMAN SECURITY APR 13, 2021 12:01 AM

100 Million More IoT Devices Are Exposed—and They Won't Be the Last

The Name:Wreck flaws in TCP/IP are the latest in a series of vulnerabilities with global implications.



Shodan 2000

Explore the Internet in style using an 80's retro-futuristic interface to synthwave music.

Webcam

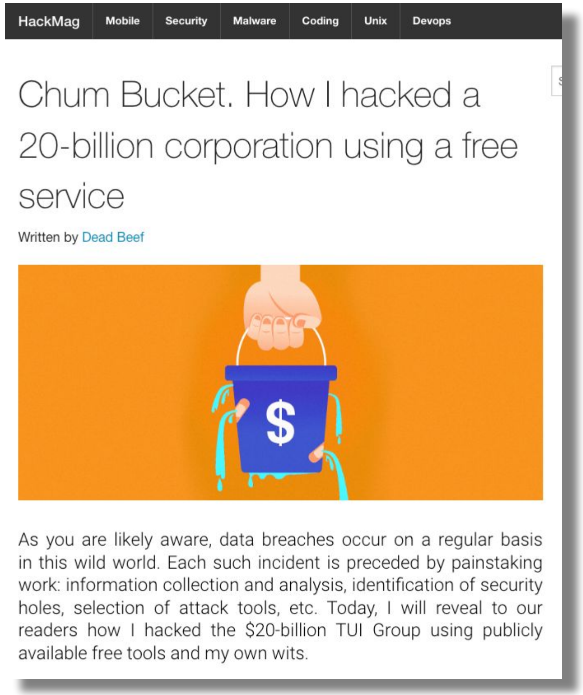
best ip cam search I have found yet.

▲ 12,519 webcam surveillance cams

Cams

admin admin

▲ 5,290 cam webcam



Chum Bucket. How I hacked a 20-billion corporation using a free service

Written by Dead Beef

As you are likely aware, data breaches occur on a regular basis in this wild world. Each such incident is preceded by painstaking work: information collection and analysis, identification of security holes, selection of attack tools, etc. Today, I will reveal to our readers how I hacked the $20-billion TUI Group using publicly available free tools and my own wits.

Mitigating Edge security risk

Red Hat platforms for the edge

Small footprint edge OS

Memory-constrained edge servers/Internet of Things (IoT) Gateways

▸ Today

Single-node edge servers

Low bandwidth or disconnected sites

▸ Today

Remote worker nodes

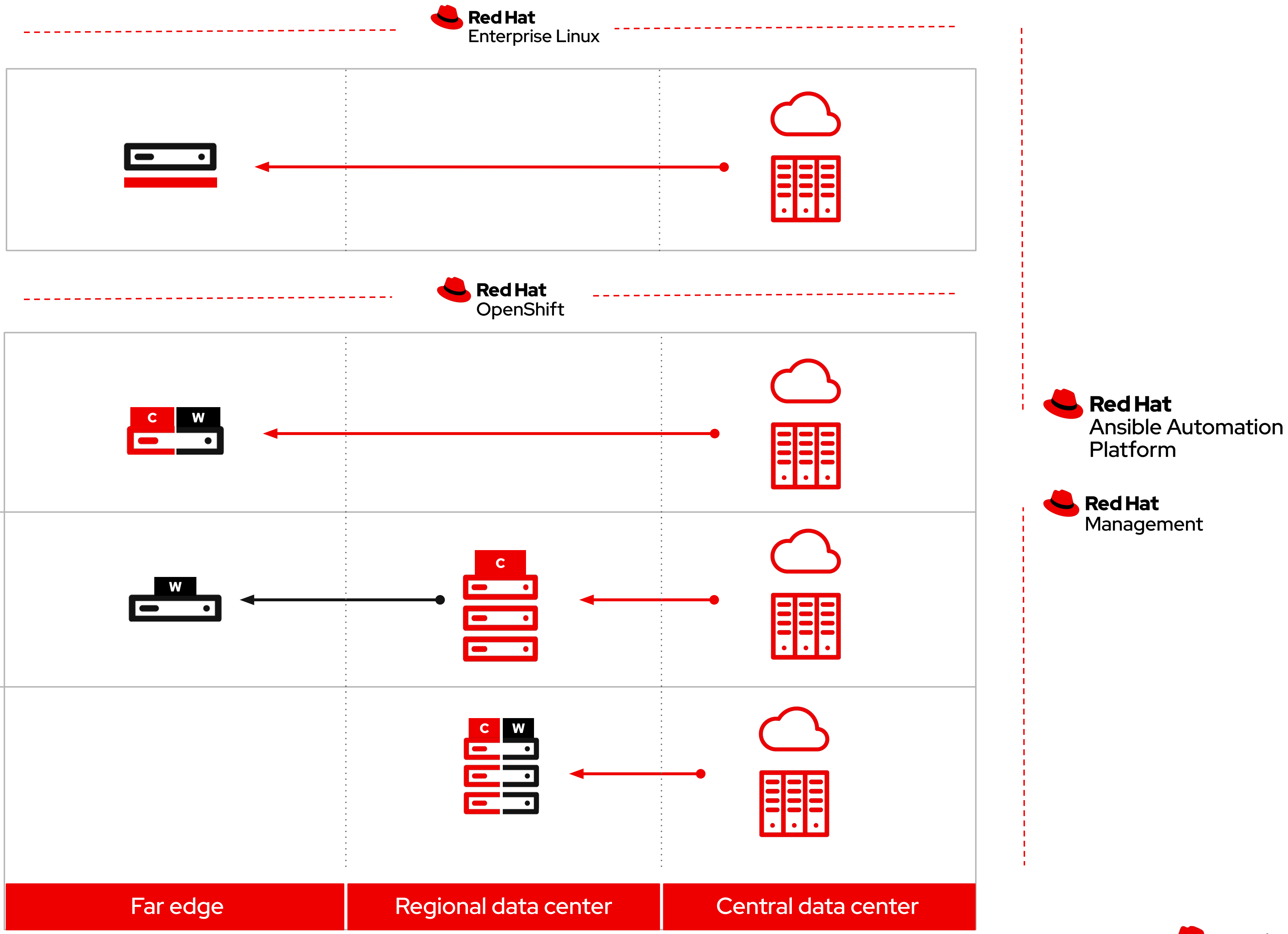
Space-constrained environments

▸ Today

3 node Clusters

Small footprint with high availability

▸ Today



← Cluster management and application deployment   ← Kubernetes node control   C Control node   W Worker node

Edge computing with Red Hat Enterprise Linux

Ensured stability and deployment flexibility

Quick image generation

Efficiently create purpose-built operating system (OS) images optimized for the architectural challenges inherent at edge locations

Edge management

Improve security and scale with the benefits of zero-touch provisioning, fleet health visibility, and quick security remediations throughout the entire life cycle

Efficient over-the-air updates

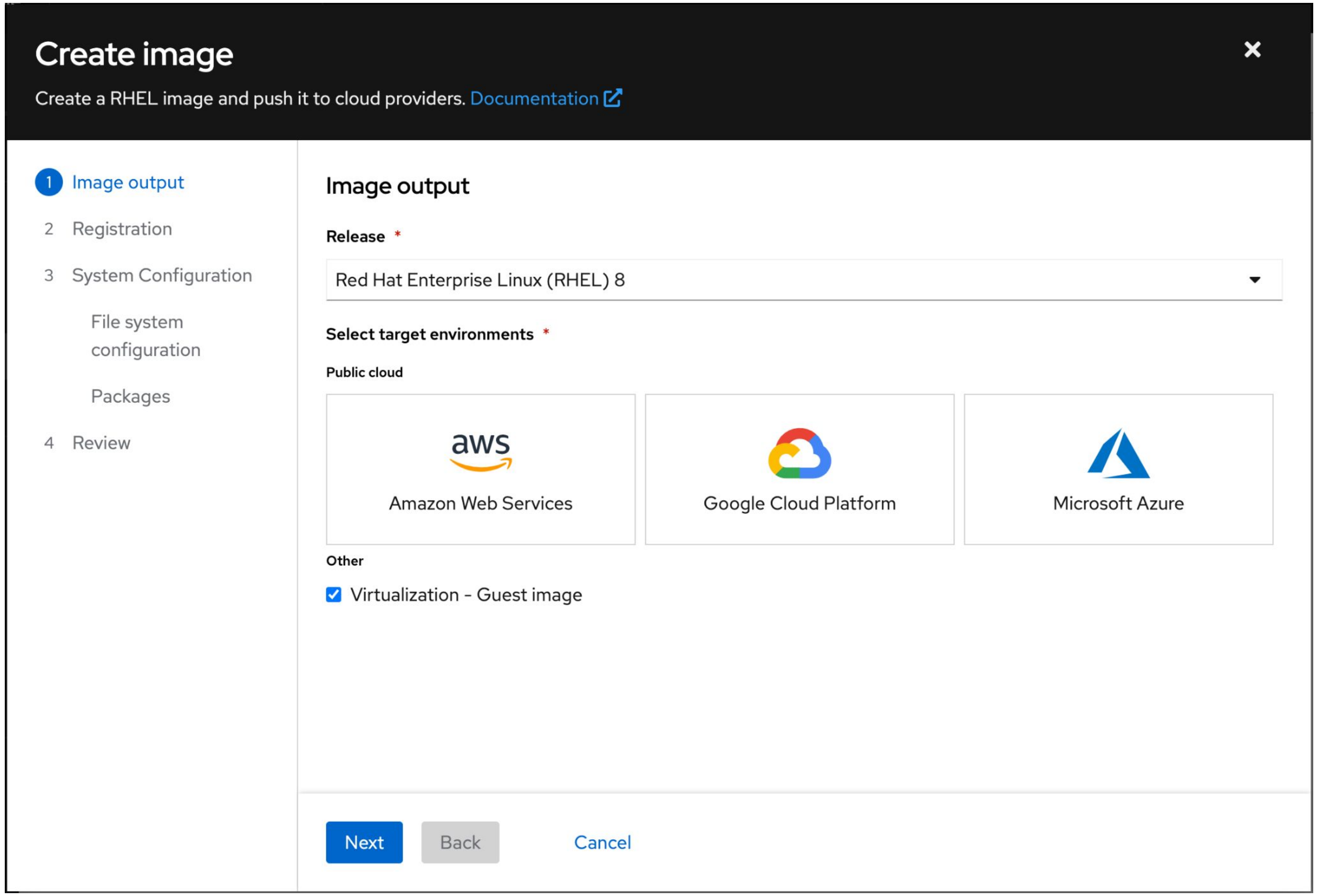
Updates transfer significantly less data and are ideal for remote sites with limited or intermittent connectivity

Intelligent rollbacks

Application-specific health checks detect conflicts and automatically revert an OS update, preventing downtime

Red Hat Enterprise Linux image builder

Save time and ensure consistency when deploying RHEL systems at scale



Create image

Create a RHEL image and push it to cloud providers. Documentation

1 Image output
2 Registration
3 System Configuration
File system configuration
Packages
4 Review

Image output

Release *

Red Hat Enterprise Linux (RHEL) 8

Select target environments *

Public cloud

Amazon Web Services | Google Cloud Platform | Microsoft Azure

Other

☑ Virtualization - Guest image

- **Support for Bare Metal Deployments**
  Install a customized RHEL OS image directly on physical hardware by creating installation media with a built-in kickstart file to automate the process.

- **Customized Filesystem Support**
  Assemble RHEL OS images that have multiple, distinct, non-LVM filesystem mount points rather than a single, large root filesystem.

Steps for using image builder

1. Choose platform

Physical, private cloud, public cloud, or edge

2. Select image builder tool

Image builder service
console.redhat.com

Image builder
On-premises private build

3. Create blueprint

Define and customize the image

4. Build the image

Create a variety of images including Red Hat OpenStack, Amazon Web Services, VMware, and Microsoft Azure, and more

5. Deploy instance

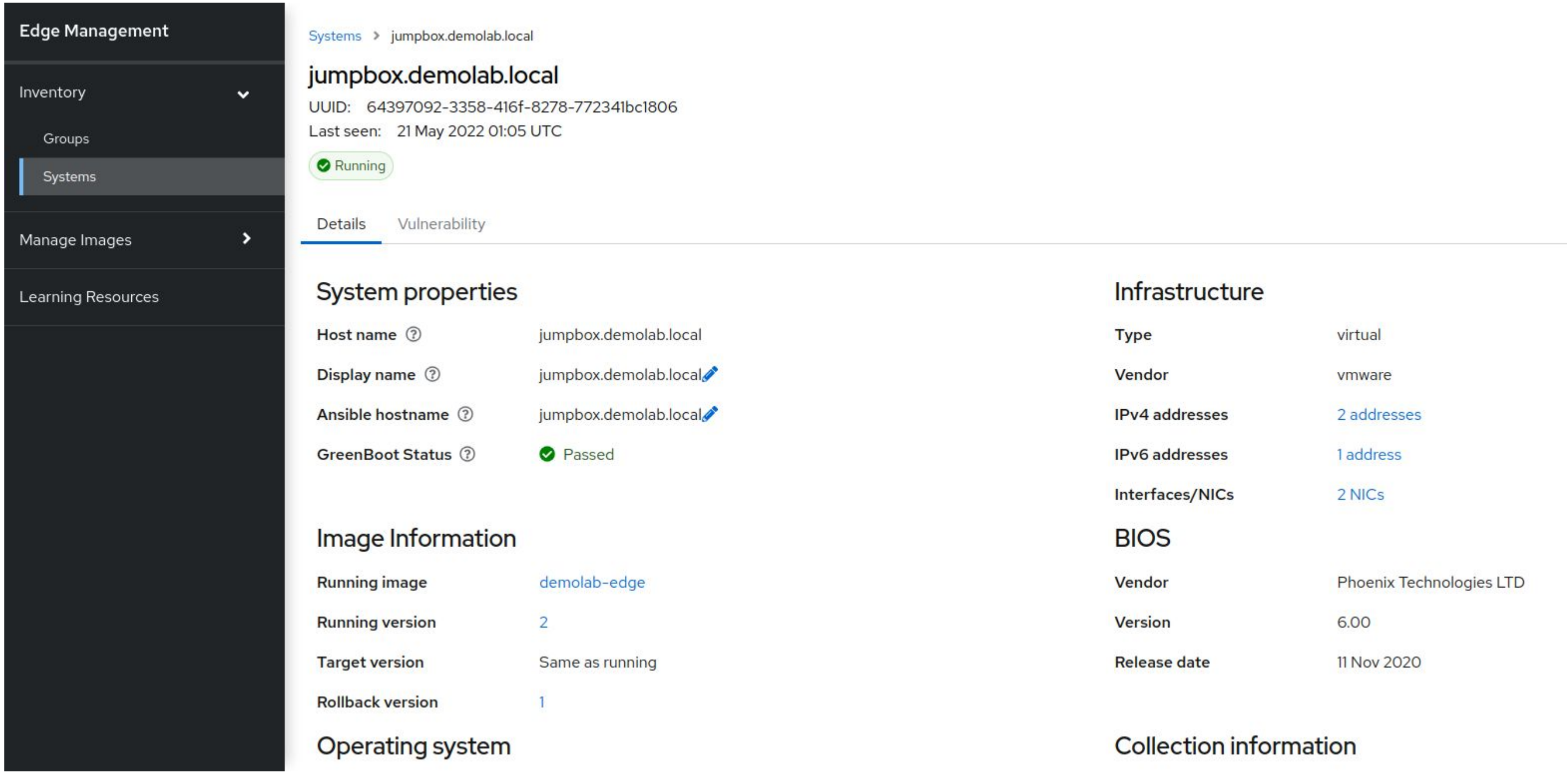
Push image to the cloud provider of your choice or download to your datacenter

DEMO: Image builder - RHEL for Edge

But wait, there is more!

Red Hat Edge Management

console.redhat.com



jumpbox.demolab.local

UUID: 64397092-3358-416f-8278-772341bc1806

Last seen: 21 May 2022 01:05 UTC

Running

System properties

Host name	jumpbox.demolab.local
Display name	jumpbox.demolab.local
Ansible hostname	jumpbox.demolab.local
GreenBoot Status	Passed

Image Information

Running image	demolab-edge
Running version	2
Target version	Same as running
Rollback version	1

Operating system

Infrastructure

Type	virtual
Vendor	vmware
IPv4 addresses	2 addresses
IPv6 addresses	1 address
Interfaces/NICs	2 NICs

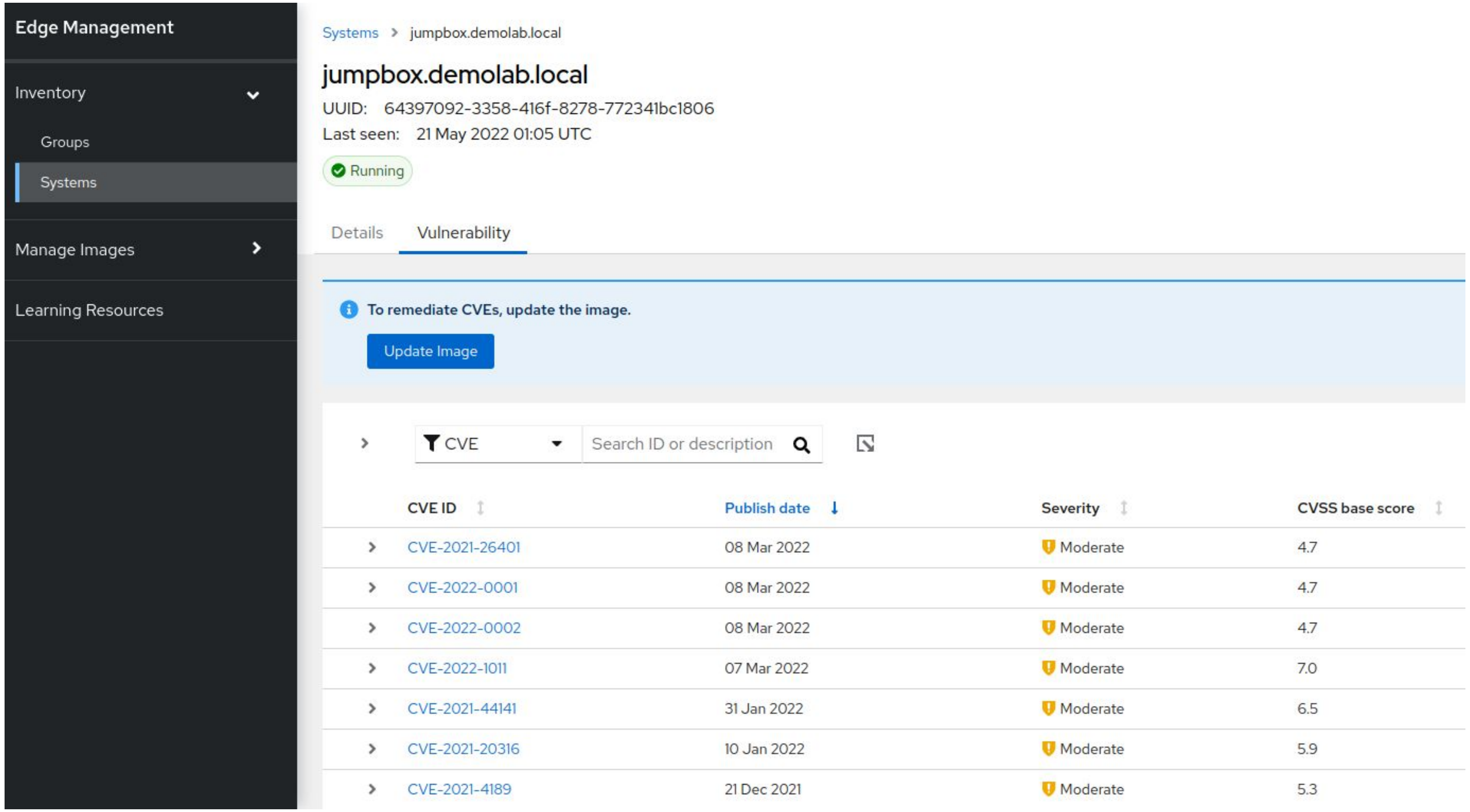
BIOS

Vendor	Phoenix Technologies LTD
Version	6.00
Release date	11 Nov 2020

Collection information

Red Hat Edge Management

console.redhat.com



jumpbox.demolab.local

UUID: 64397092-3358-416f-8278-772341bc1806

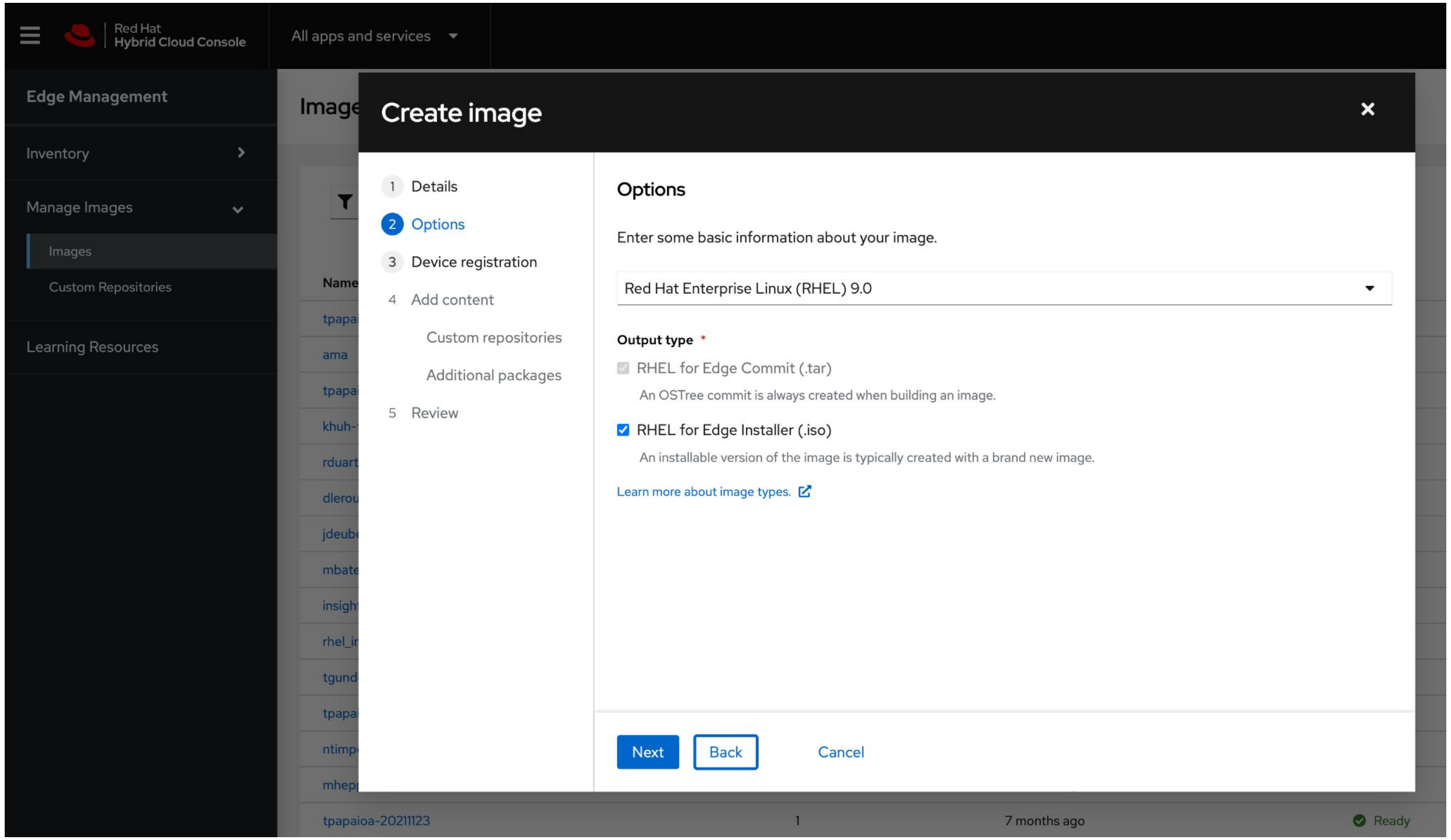
Last seen: 21 May 2022 01:05 UTC

To remediate CVEs, update the image.

CVE ID	Publish date	Severity	CVSS base score
CVE-2021-26401	08 Mar 2022	Moderate	4.7
CVE-2022-0001	08 Mar 2022	Moderate	4.7
CVE-2022-0002	08 Mar 2022	Moderate	4.7
CVE-2022-1011	07 Mar 2022	Moderate	7.0
CVE-2021-44141	31 Jan 2022	Moderate	6.5
CVE-2021-20316	10 Jan 2022	Moderate	5.9
CVE-2021-4189	21 Dec 2021	Moderate	5.3

Red Hat Edge Management

console.redhat.com



Create image

1. Details
2. Options
3. Device registration
4. Add content
   - Custom repositories
   - Additional packages
5. Review

Options

Enter some basic information about your image.

Red Hat Enterprise Linux (RHEL) 9.0

Output type *

☑ RHEL for Edge Commit (.tar)

An OSTree commit is always created when building an image.

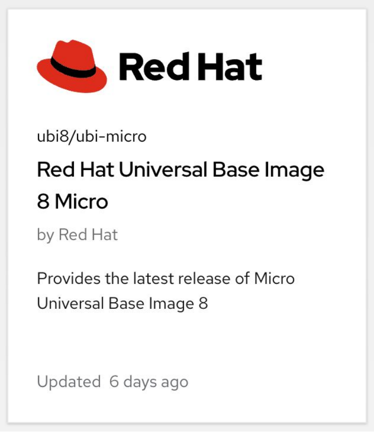
☑ RHEL for Edge Installer (.iso)

An installable version of the image is typically created with a brand new image.

Learn more about image types.

DEMO: Container deployment to the Edge

Running containers on RHEL for Edge



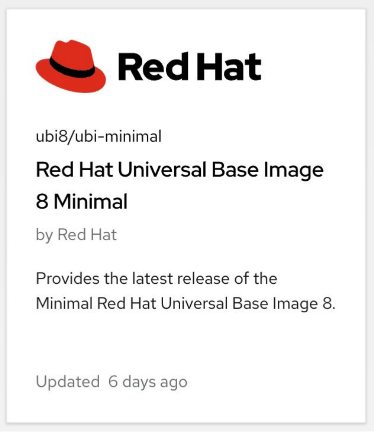
ubi8/ubi-micro

Red Hat Universal Base Image 8 Micro

by Red Hat

Provides the latest release of Micro Universal Base Image 8

Updated 6 days ago



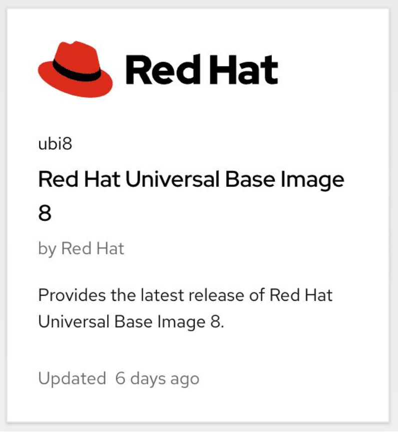
ubi8/ubi-minimal

Red Hat Universal Base Image 8 Minimal

by Red Hat

Provides the latest release of the Minimal Red Hat Universal Base Image 8.

Updated 6 days ago



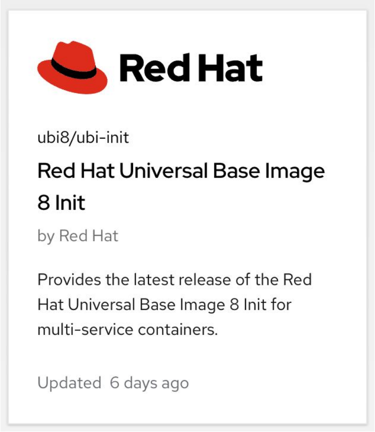
ubi8

Red Hat Universal Base Image 8

by Red Hat

Provides the latest release of Red Hat Universal Base Image 8.

Updated 6 days ago



ubi8/ubi-init

Red Hat Universal Base Image 8 Init

by Red Hat

Provides the latest release of the Red Hat Universal Base Image 8 Init for multi-service containers.

Updated 6 days ago

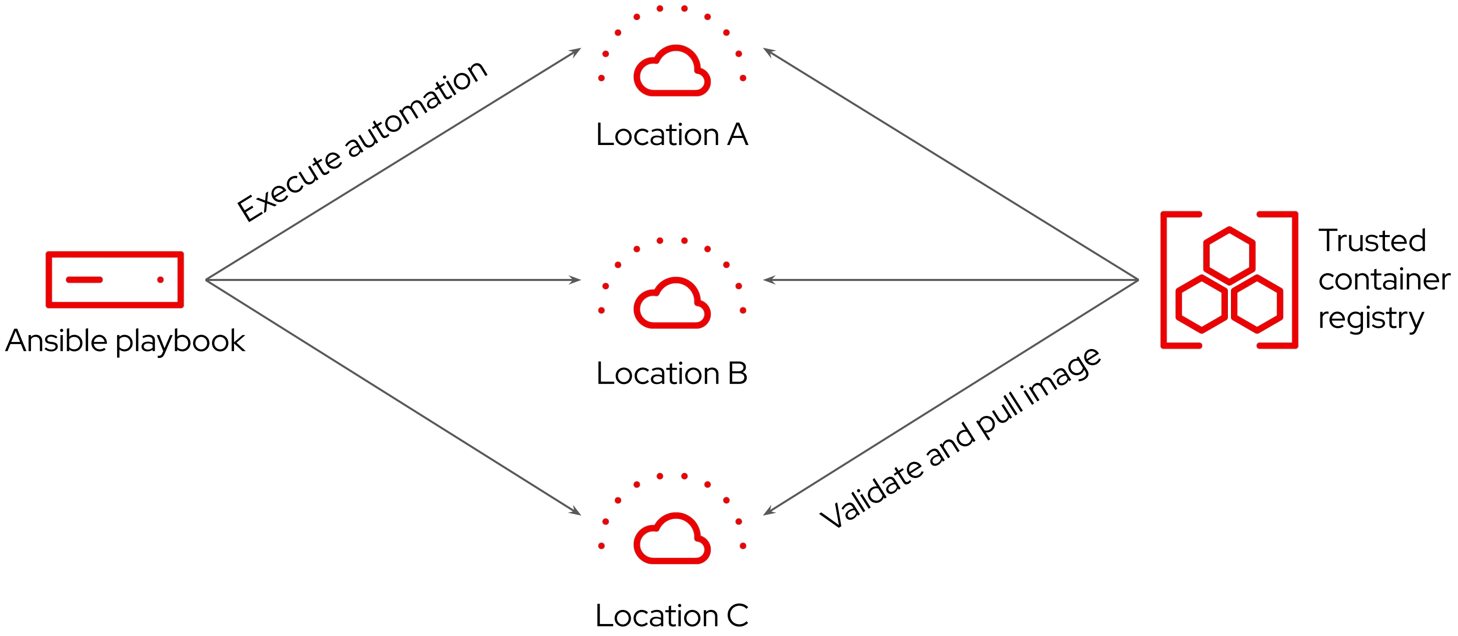
Choose image based on your requirements

podman

Red Hat
Ansible Automation
Platform

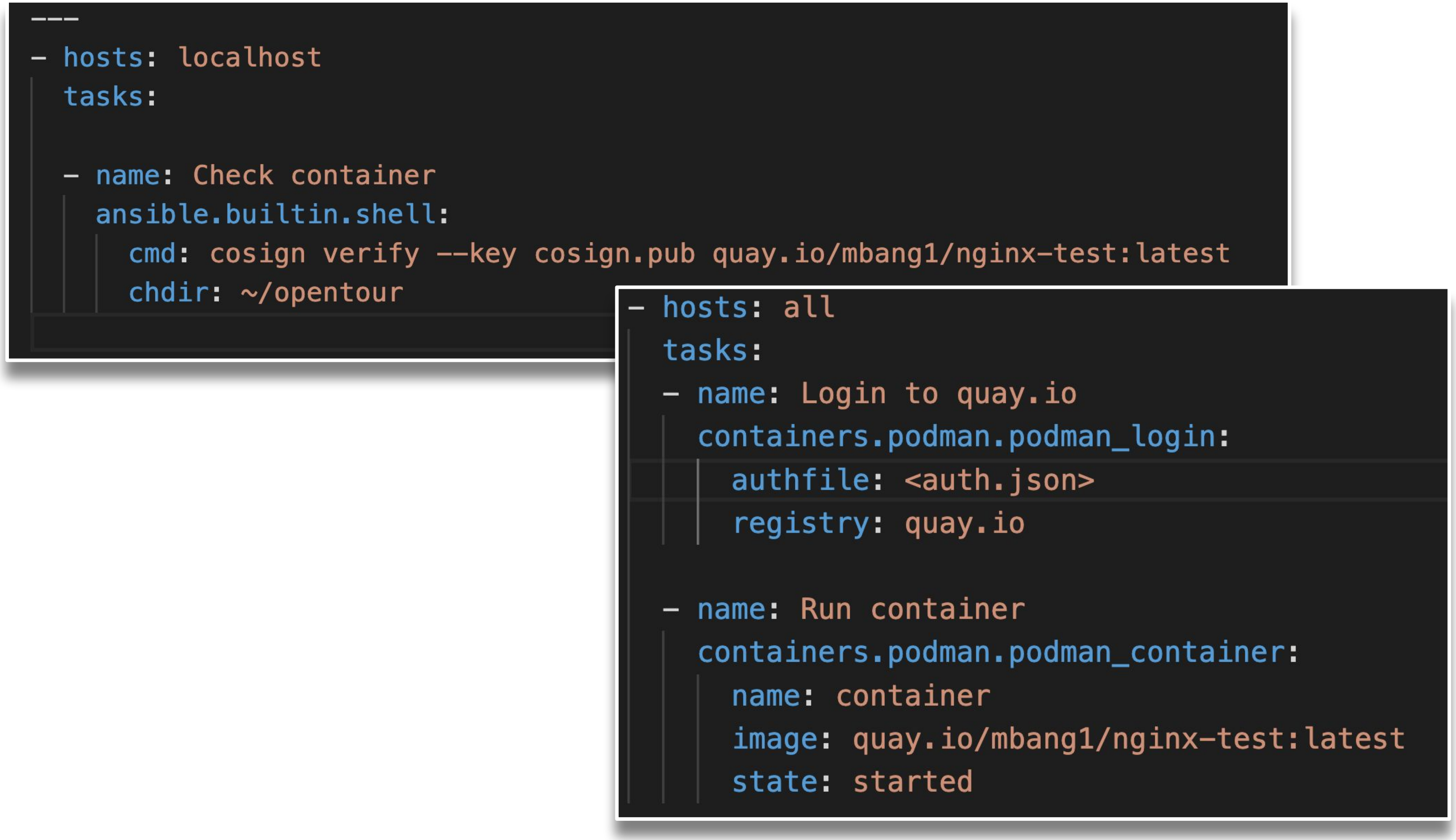
Running containers on RHEL for Edge

- Use trusted repositories (registry.redhat.io…)
- Use podman, which is designed to be secure:
  - It uses SELinux, signed images, integrates with Linux capabilities and runs as non privileged user.
- Use Ansible – Can deploy containers to many edge servers
  - Scalable and consistent
  - Allows you to reuse processes from your core data center(s)



Ansible is used to orchestrate deployment of containers that are check for valid signing before being run

Example playbooks



```yaml
---
- hosts: localhost
  tasks:

  - name: Check container
    ansible.builtin.shell:
      cmd: cosign verify --key cosign.pub quay.io/mbang1/nginx-test:latest
      chdir: ~/opentour
```

```yaml
- hosts: all
  tasks:
  - name: Login to quay.io
    containers.podman.podman_login:
      authfile: <auth.json>
      registry: quay.io

  - name: Run container
    containers.podman.podman_container:
      name: container
      image: quay.io/mbang1/nginx-test:latest
      state: started
```

opentour — mbang@localhost:~ — ssh mbang@localhost -p 5022 — 96×33

```
[mbang@localhost ~]$ 
```

opentour — mbang@localhost:~ — ssh mbang@localhost -p 5022 — 96×33

```
[mbang@localhost ~]$ 
```

Key takeaways

- Next time you download something from Internet, think twice
- Sign & verify must be a mandatory requirement
- Don’t turn GPGCheck off
- Don’t use latest tag
- Choose your container base image wisely
- Use trusted repositories
- Let SELinux be enforcing
- You **have** to manage edge devices and do it easily

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

- linkedin.com/company/red-hat
- youtube.com/user/RedHatVideos
- facebook.com/redhatinc
- twitter.com/RedHat