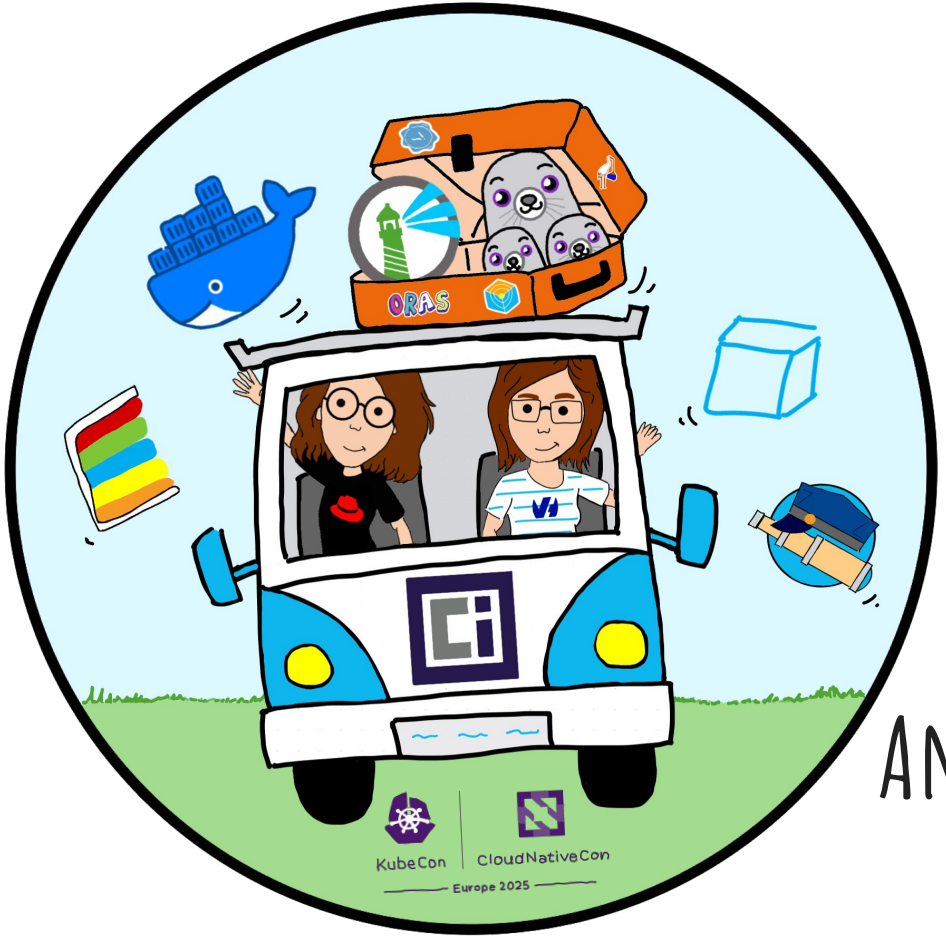


The Ultimate Container Challenge

An interactive trivia game on OCI, podman, docker...

Aurélie Vache - Sherine Khoury

Aurélie Vache

@aurelievache

Developer Advocate at OVHcloud

💙 Deaf & Hard of Hearing WG

❤ Open Source

📚 Tech writer & Book author

✏️ Sketchnoter & ❤ Retrogaming

https://developers.events

https://www.youtube.com/AurelieVache

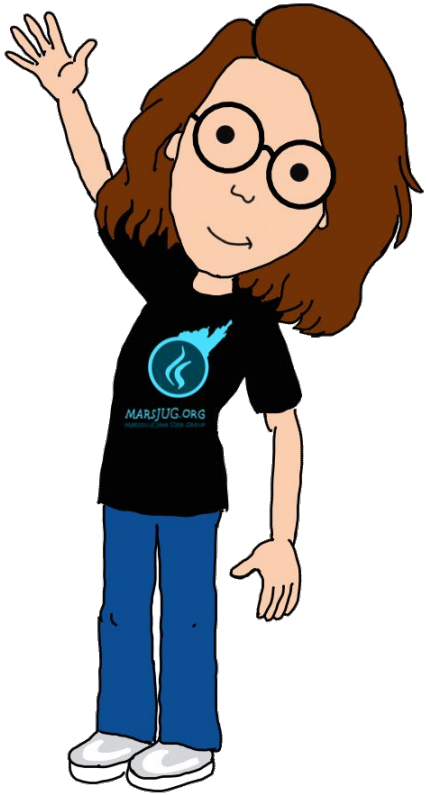
Sherine Khoury

@srinerine

Red Hat Senior Software Engineer

Marseille JUG community member

Jogger!



The ultimate container challenge > rules

KubeCon CloudNativeCon Europe 2025

Connect to Slido: slido.com

Code: **1835927**

Fill in your alias at Question 1

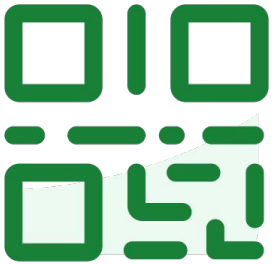
Your goal: learn, play & have fun.

🎯 Give correct answers as fast as you can

🎁 To win interesting prizes!

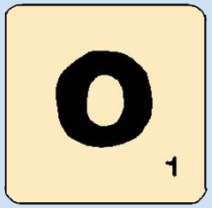
GO, GO, GO!

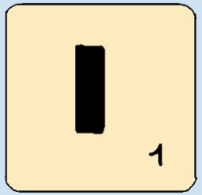
slido

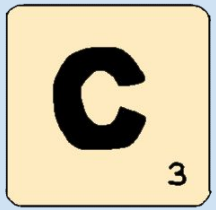


Join at slido.com
#1835927

ⓘ Click **Present with Slido** or install our Chrome extension to display joining instructions for participants while presenting.







Question n°1

slido

What is OCI?

ⓘ Click **Present with Slido** or install our Chrome extension to activate this poll while presenting.

What is OCI?

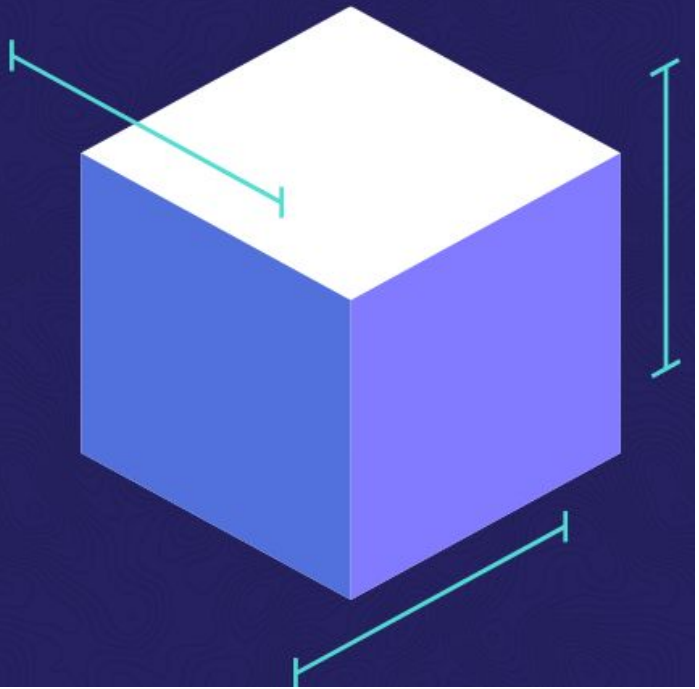
A- Optimal Container Initiative
B- Open Container Image
C- Open Container Initiative
D- Open CloudNative Initiative

Open Container Initiative

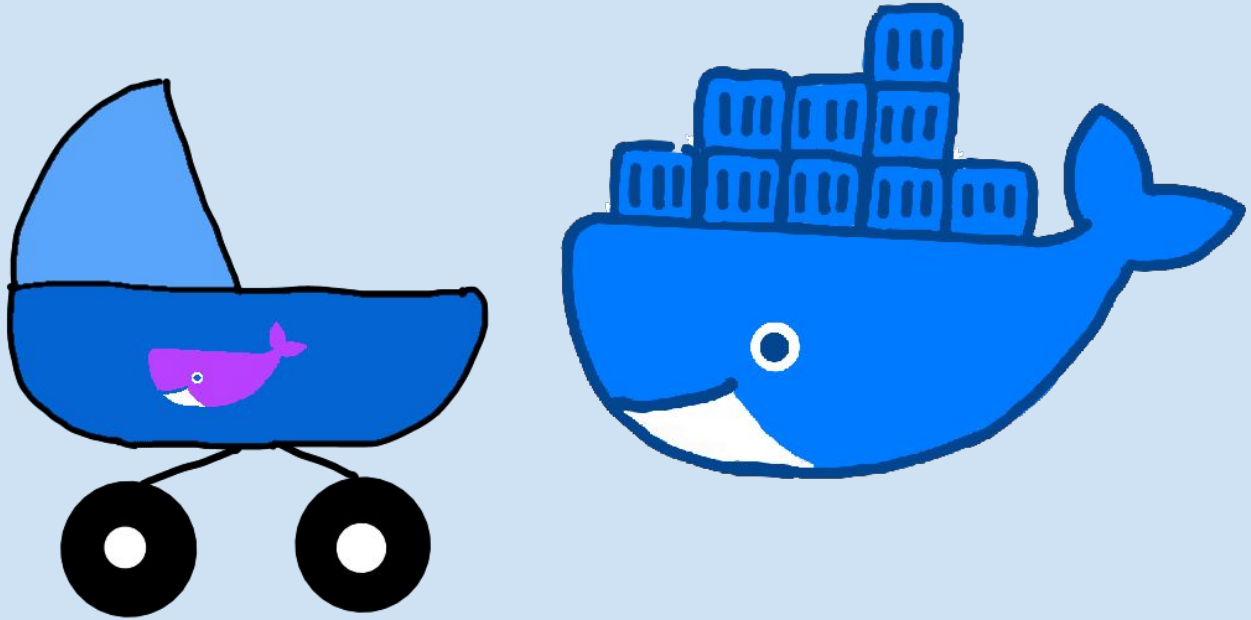
Open Container Initiative

The **Open Container Initiative** is an open governance structure for the express purpose of creating open industry standards around container formats and runtimes.

Established in June 2015 by Docker and other leaders in the container industry, the OCI currently contains three specifications: the Runtime Specification (runtime-spec), the Image Specification (image-spec) and the Distribution Specification (distribution-spec). The Runtime Specification outlines how to run a “filesystem bundle” that is unpacked on disk. At a high-level an OCI implementation would download an OCI Image then unpack that image into an OCI Runtime filesystem bundle. At this point the OCI Runtime Bundle would be run by an OCI Runtime.



https://opencontainers.org/



Question n°2

slido

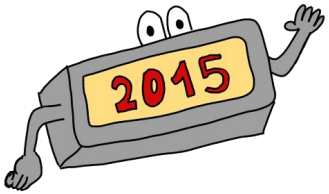
So OCI & Docker are the same thing?

ⓘ Click **Present with Slido** or install our Chrome extension to activate this poll while presenting.

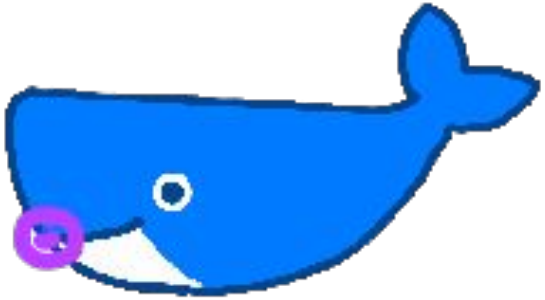
So OCI & Docker are the same thing?

A- True
B- False

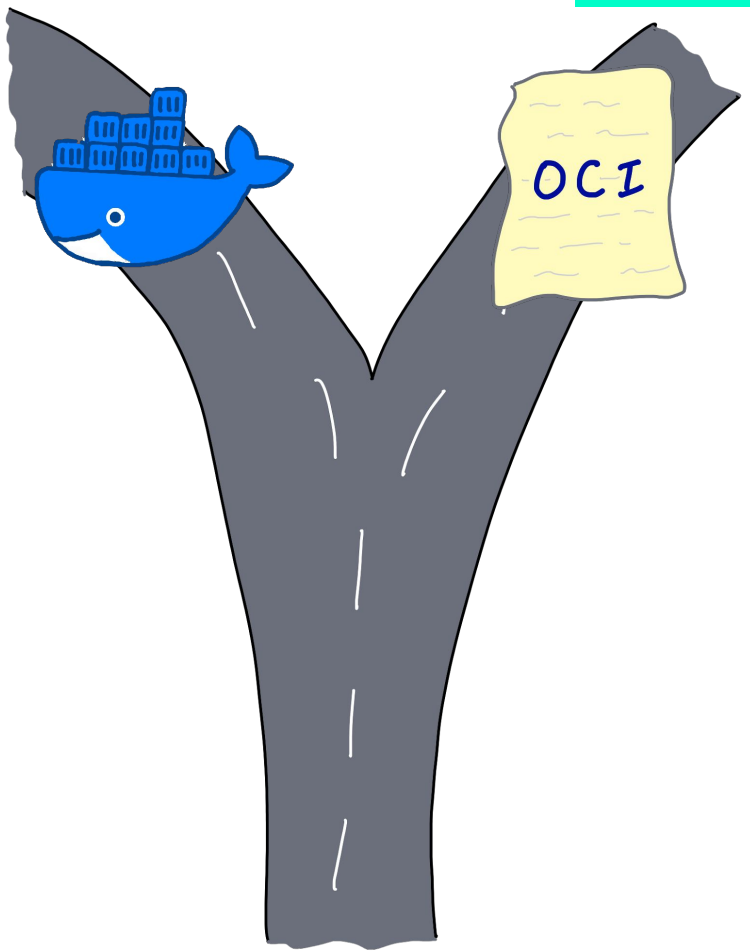
OCI = Docker's OpenSource baby 👶



Docker donated its image format and its runtime to OCI 🎉

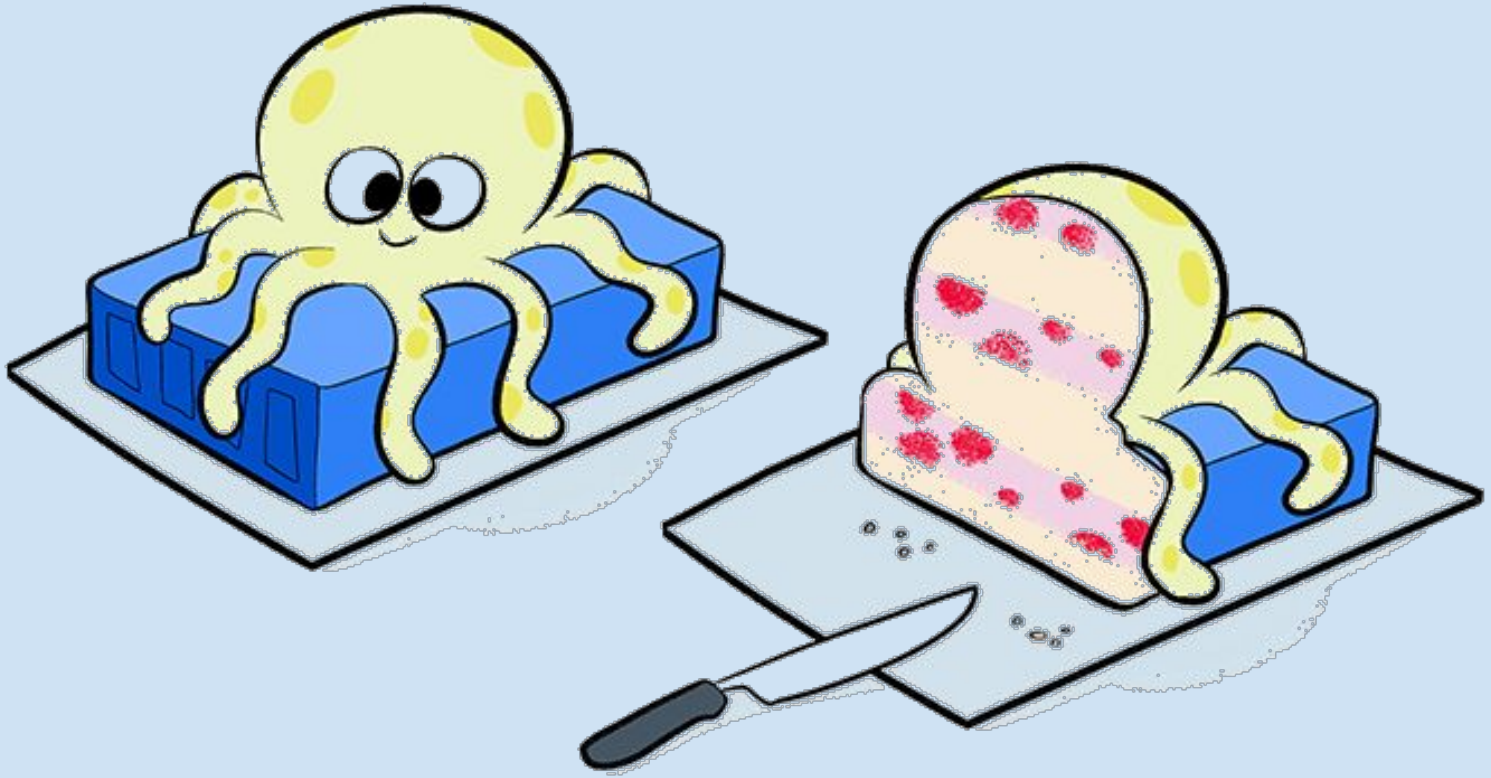


OCI = Docker's OpenSource baby



Since, OCI & Docker evolve separately.

A few subtle differences exist between them,
but (maintainers of) registries and CLIs do their best to hide these differences for a better user experience.



Question n°3

slido

Is Docker image format identical to OCI's?

ⓘ Click **Present with Slido** or install our Chrome extension to activate this poll while presenting.

Is Docker image format identical to OCI's?

A- OCI & DockerV2: same thing!
B- OCI is a fork of DockerV1
C- OCI & DockerV2 are slightly different
D- OCI & DockerV2 are totally different

Demo!

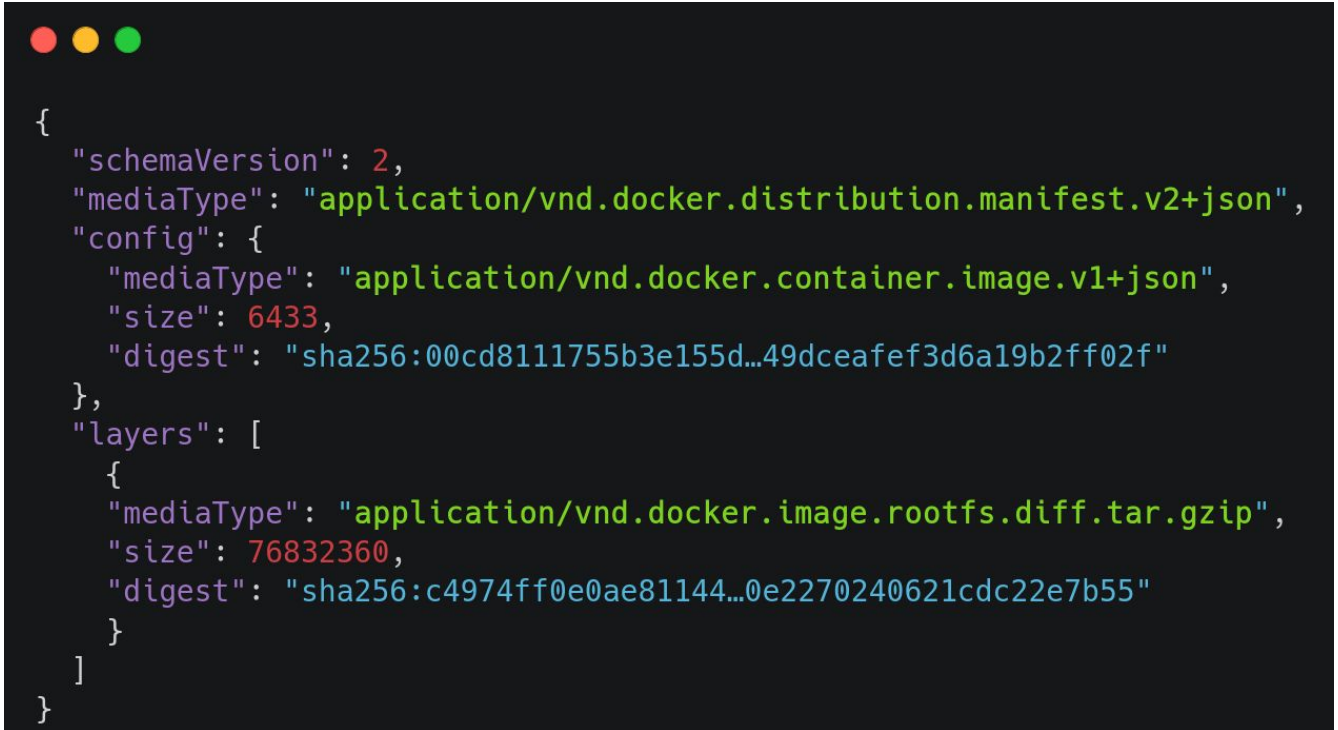
Image formats: OCI ≠ Docker (compatibilities)

OCI

```
{
  "schemaVersion": 2,
  "mediaType": "application/vnd.oci.image.manifest.v1+json",
  "config": {
    "mediaType": "application/vnd.oci.image.config.v1+json",
    "digest": "sha256:663acf02e20300de…c90b4c11887648f15a22a0",
    "size": 22046
  },
  "layers": [
    {
    "mediaType": "application/vnd.oci.image.layer.v1.tar+gzip",
    "digest": "sha256:54af59ad79cd269b…16eddfaff1284a2d30957e",
    "size": 80309925
    }
  ]
}
```

Image formats: OCI ≠ Docker (compatibilities)

Docker



```
{
  "schemaVersion": 2,
  "mediaType": "application/vnd.docker.distribution.manifest.v2+json",
  "config": {
    "mediaType": "application/vnd.docker.container.image.v1+json",
    "size": 6433,
    "digest": "sha256:00cd8111755b3e155d…49dceafef3d6a19b2ff02f"
  },
  "layers": [
    {
    "mediaType": "application/vnd.docker.image.rootfs.diff.tar.gzip",
    "size": 76832360,
    "digest": "sha256:c4974ff0e0ae81144…0e2270240621cdc22e7b55"
    }
  ]
}
```

MANIFEST INCOMPATIBILITIES

application/vnd.oci.image.manifest.v1+json

Similar/related schema:

- application/vnd.docker.distribution.manifest.v2+json
  - **.annotations**: only present in OCI
  - **.config.annotations**: only present in OCI
  - **.config.urls**: only present in OCI
  - **.[]layers.annotations**: only present in OCI

Source:
https://github.com/opencontainers/image-spec/blob/main/media-types.md#compatibility-matrix

Config layer Incompatibilities

application/vnd.oci.image.config.v1+json

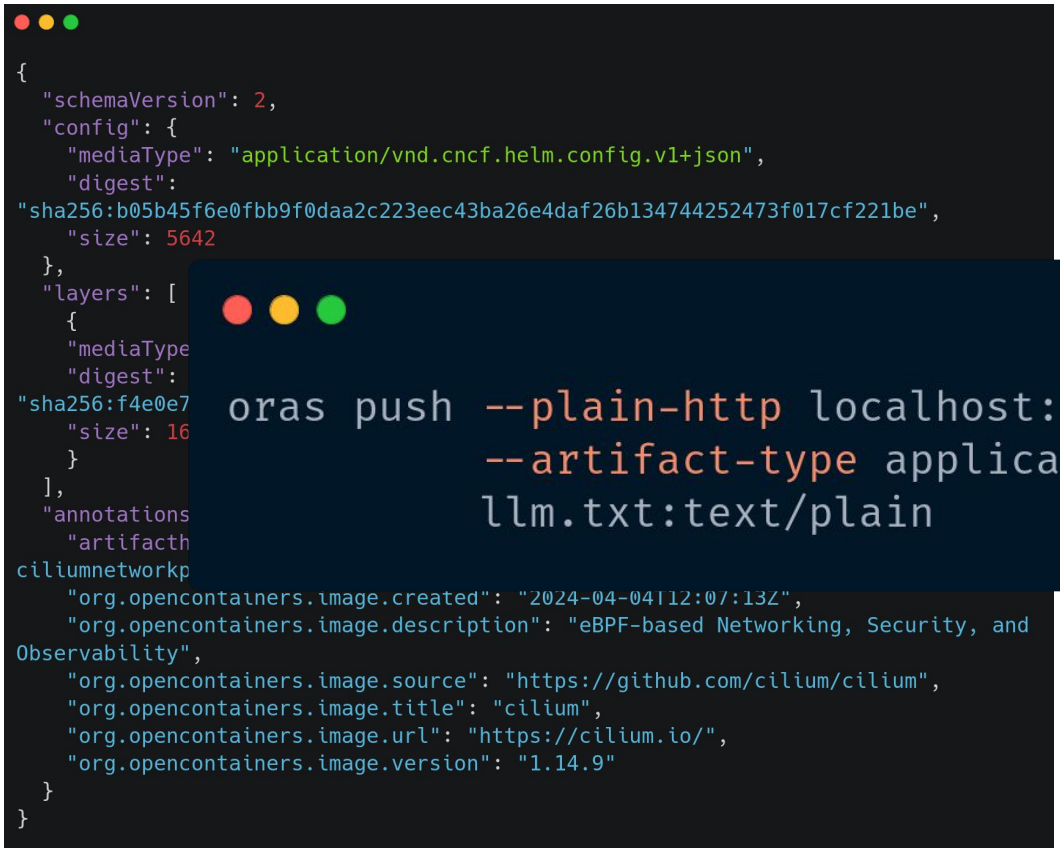
Similar/related schema:

- application/vnd.docker.container.image.v1+json (Docker Image Spec v1.2)
  - **.config.Memory**: only present in Docker, and reserved in OCI
  - **.config.MemorySwap**: only present in Docker, and reserved in OCI
  - **.config.CpuShares**: only present in Docker, and reserved in OCI
  - **.config.Healthcheck**: only present in Docker, and reserved in OCI

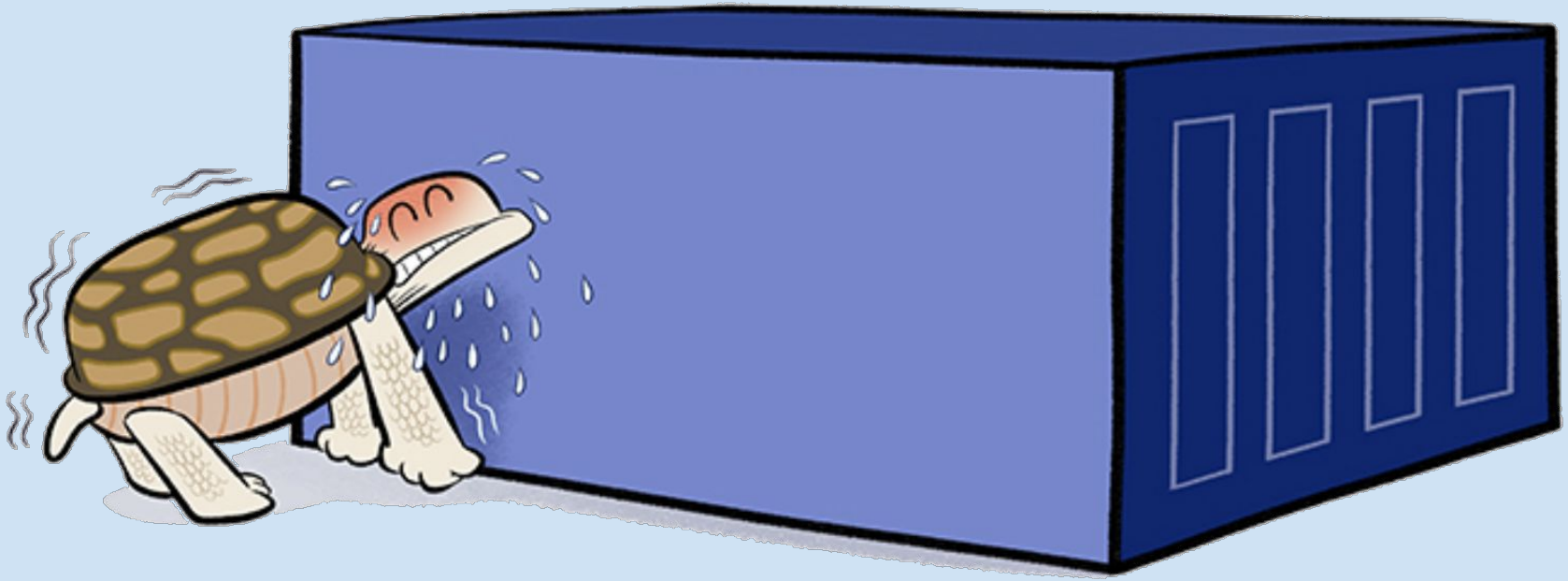
New Possibilities: Helm, Your own artifacts... LLM models tomorrow?


```
{
  "schemaVersion": 2,
  "config": {
    "mediaType": "application/vnd.cncf.helm.config.v1+json",
    "digest":
"sha256:b05b45f6e0fbb9f0daa2c223eec43ba26e4daf26b134744252473f017cf221be",
    "size": 5642
  },
  "layers": [
    {
    "mediaType": "application/vnd.cncf.helm.chart.content.v1.tar+gzip",
    "digest":
"sha256:f4e0e7c9b8dd9de175a343fd4af575374e5ed4ea2e05b9299e5cd18b7b801b11",
    "size": 165199
    }
  ],
  "annotations": {
```

New Possibilities: Helm, Your own artifacts... LLM models tomorrow?




```
oras push --plain-http localhost:5000/hello-artifact:v1 \
          --artifact-type application/vnd.sunny.tech.config \
          llm.txt:text/plain
```



Question n°4

slido

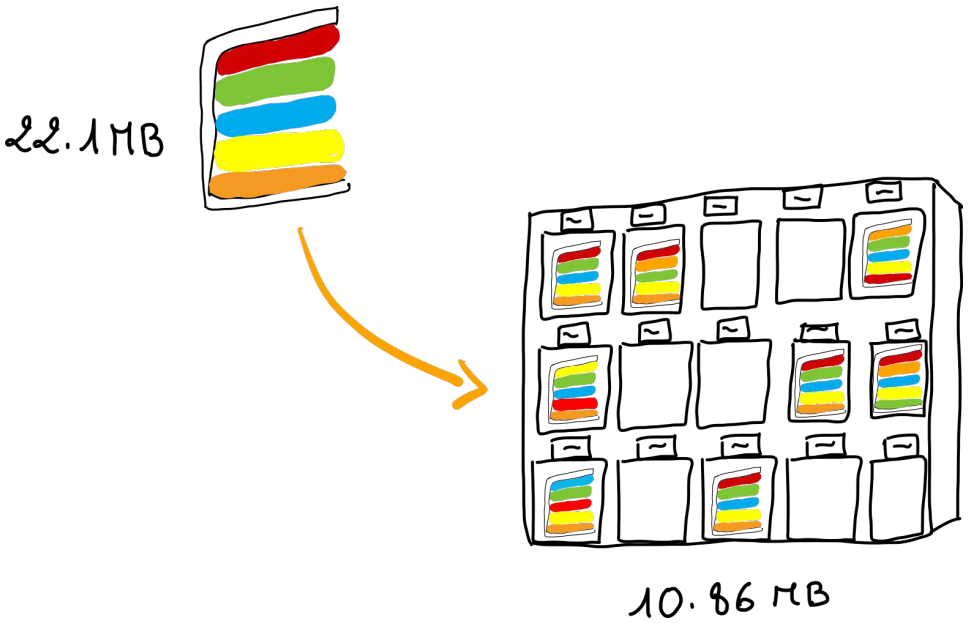
What happens when I push an image to a registry?

ⓘ Click **Present with Slido** or install our Chrome extension to activate this poll while presenting.

What happens when I push my image to a registry?

A- My image is pushed as is
B- My image is compressed upon push
C- Each registry has its own compression algorithm

Oh yes, size matters! ^^

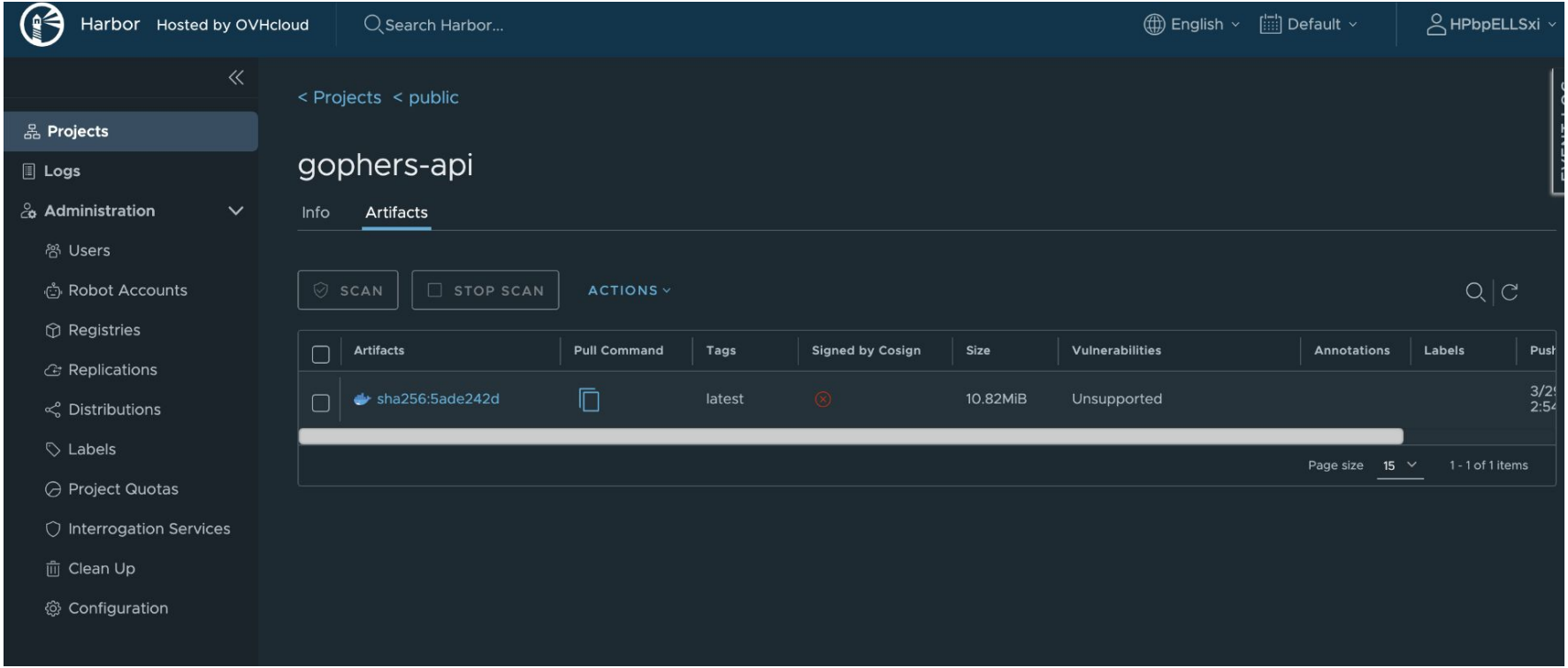


- The image size on the local machine and on the registry doesn't match.
- When the image is pushed, the client compresses the layers on the fly

Demo!

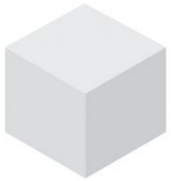
Demo with Docker

```
$ docker build -t cx6ds30d.gra7.container-registry.ovh.net/public/gophers-api
.

$ docker image ls
REPOSITORY                                                TAG     IMAGE ID    CREATED             SIZE
cx6ds30d.gra7.container-registry.ovh.net/public/gophers-api   latest    b3a1bc8451d3   About a minute ago   22.6MB

 $ docker push cx6ds30d.gra7.container-registry.ovh.net/public/gophers-api
```



22.6MB en local → 10.82MiB sur OVHcloud Managed Private Registry / Harbor ;-)

https://79352h8v.c1.de1.container-registry.ovh.net



scraly/gophers-api ☆0

By scraly • Updated about 5 hours ago

Image

Pulls 63

Overview **Tags**

TAG

latest

Last pushed **8 days ago** by scraly

`docker pull scraly/gophers-api:latest` Copy

Digest	OS/ARCH	Compressed Size
088713707029	linux/amd64	10.82 MB

10.82MiB aussi sur DockerHub ;-)

https://hub.docker.com/r/scraly/gophers-api/tags

Et lorsque l’on pull une image, les layers sont décompressées, l’image a de nouveau sa taille d’origine :

```
$ docker pull cx6ds30d.gra7.container-registry.ovh.net/public/gophers-api

$ docker image ls
REPOSITORY                                                TAG     IMAGE ID    CREATED         SIZE
cx6ds30d.gra7.container-registry.ovh.net/public/gophers-api   latest    b3a1bc8451d3   11 minutes ago   22.6MB
```

Test With Podman

```
$ podman image ls
REPOSITORY                      TAG         IMAGE ID      CREATED         SIZE
…
docker.io/scraly/gophers-api    podman      18d2332b76cb  37 minutes ago  24
MB

$ podman push docker.io/scraly/gophers-api:podman
```

Test With Podman

TAG

podman

Last pushed **a few seconds ago** by scraly

```
docker pull scraly/gophers-api:podman
```
Copy

Digest	OS/ARCH	Compressed Size
b3e6777679b6	linux/amd64	10.82 MB

https://hub.docker.com/r/scraly/gophers-api/tags

On peut pusher une image en changeant l’algo de compression :

```
$ podman tag 18d2332b76cb docker.io/scraly/gophers-api:podman-zstd

$ podman push docker.io/scraly/gophers-api:podman-zstd --compression-
format=zstd
```

TAG

podman-zstd

Last pushed **a few seconds ago** by scraly

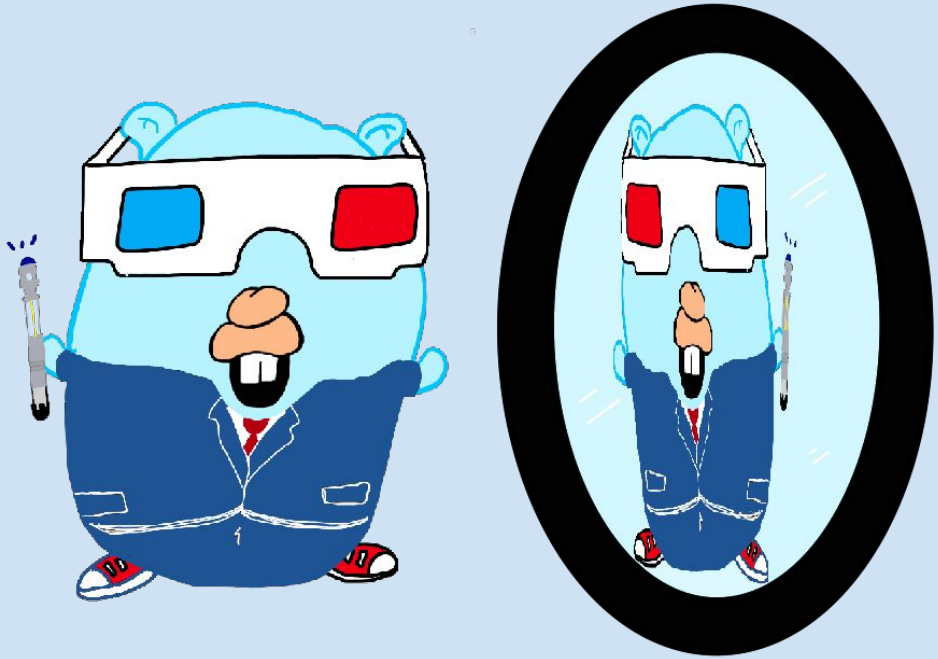
```
docker pull scraly/gophers-api:podman-zstd
```
Copy

Digest	OS/ARCH	Compressed Size
60f3676696ba	amd64	10.6 MB

https://hub.docker.com/r/scraly/gophers-api/tags

⚙️ Which Algorithm to Choose?

Algorithm	Speed (Compression/Decompression)	Compression Rate	Ideal for
`gzip`	Fast / Very fast	Medium	General use
`xz`	Slow / Medium	Very good	Long-term archives
`zstd`	Very fast / Fast	Excellent	Large volumes, CI/CD
`uncompressed`	Instantaneous	N/A	Quick tests



Question n°5

slido

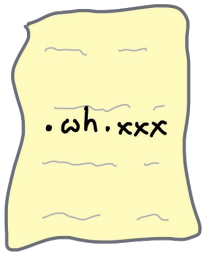
When using `RUN rm -r /helm` in a Dockerfile to remove a folder

When using `RUN rm -r /helm` in a Dockerfile to remove a folder

A- Image size decreases
B- Image size stays the same
C- Image size is almost the same
D- Image size increases

The size of the image is almost the same

- Nothing is lost, nothing is created, everything is transformed!! (LaVoisier)
  - A new layer is added.
  - It contains a single file: **.wh.**helm (0bytes)
  - No files are really deleted
  - The container runtime will delete **/helm** upon loading and running the image



Demo!


$ cat Containerfile
FROM registry.access.redhat.com/ubi9/ubi-micro:latest
COPY helm /helm
WORKDIR /
RUN rm -r /helm
$ podman build .
STEP 1/4: FROM registry.access.redhat.com/ubi9/ubi-micro:latest
STEP 2/4: COPY helm /helm
--> Using cache 1f6cea500ec12456fcde8a97b9b37209e473f5aea1c0aef693d456300e768884
--> 1f6cea500ec1
STEP 3/4: WORKDIR /
--> Using cache ed69a37f4529bbf18dc2ba1138b14e6debf49497f283ecadeee1d95ff63f8695
--> ed69a37f4529
STEP 4/4: RUN rm -r /helm
--> Using cache b7d01144ad4a48ddc110da3709214f5f6371b18f3a193540a6848c27026cb1a9
--> b7d01144ad4a
b7d01144ad4a48ddc110da3709214f5f6371b18f3a193540a6848c27026cb1a9
$ podman save --format docker-dir -o wh_test/
b7d01144ad4a48ddc110da3709214f5f6371b18f3a193540a6848c27026cb1a9
Copying blob b2cd69bd1f08 done
Copying blob 8a2e666d7324 done
Copying blob f6e375c0d5ad done
Copying config b7d01144ad done
Writing manifest to image destination
$ cd wh_test/
$ for i in `ls` ; do tar tvf $i | grep helm; done
drwxr-xr-x 0/0 0 2024-04-16 07:29 helm/
-rw-r--r-- 0/0 5642 2024-04-06 15:26 helm/
b05b45f6e0fbb9f0daa2c223eec43ba26e4daf26b134744252473f017cf221be
-rw-r--r-- 0/0 165199 2024-04-06 15:26 helm/
f4e0e7c9b8dd9de175a343fd4af575374e5ed4ea2e05b9299e5cd18b7b801b11
-rw-r--r-- 0/0 5915 2024-04-06 15:26 helm/manifest.json
-rw-r--r-- 0/0 33 2024-04-06 15:26 helm/version
---------- 0/0 0 2024-04-16 07:29 .wh.helm

Question n°6

slido

Images built locally run everywhere?

ⓘ Click **Present with Slido** or install our Chrome extension to activate this poll while presenting.

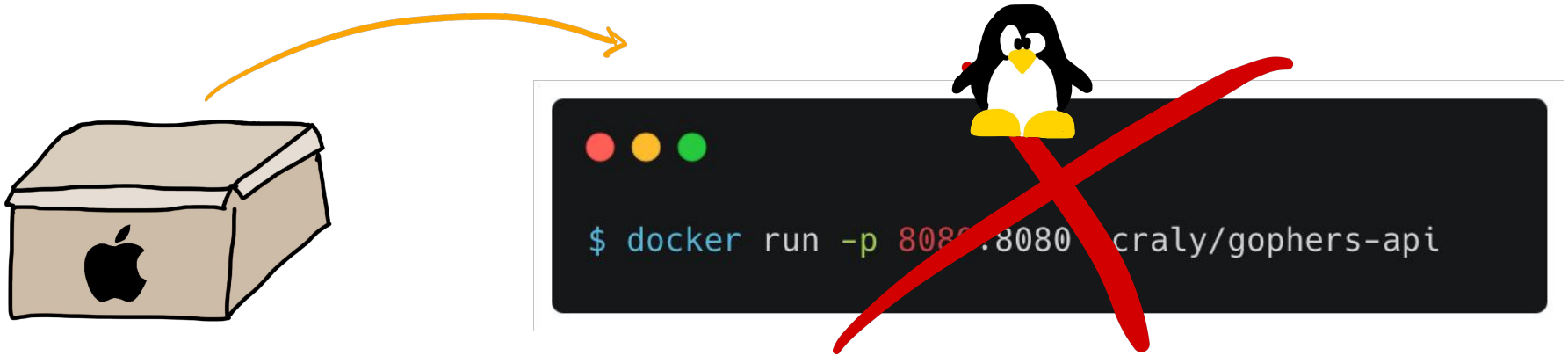
Images built locally run everywhere?

A- Obviously! “build once, deploy anywhere”
B- It depends…
C- No

“Build once, deploy anywhere”…
Could be, but beware of some pitfalls ^^

Build once, deploy anywhere ... not by default

By default, an image is built for the machine's platform and architecture.



MULTI-ARCH IMAGE TO THE RESCUE!

Demo!

1. Create a builder

```
# Create a builder to build multi-arch Docker images, and use it
$ docker buildx create --name mybuilder --bootstrap --use
```

2. INSPECT THE BUILDER

```
$ docker buildx inspect mybuilder
Name:          mybuilder
Driver:        docker-container
Last Activity: 2024-04-07 10:00:01 +0000 UTC

Nodes:
Name:      mybuilder0
Endpoint:  desktop-linux
Status:    running
Buildkit:  v0.12.4
Platforms: linux/arm64, linux/amd64, linux/amd64/v2, linux/riscv64, linux/ppc64le, linux/s390x, linux/386,
linux/mips64le, linux/mips64, linux/arm/v7, linux/arm/v6
Labels:
 org.mobyproject.buildkit.worker.executor:         oci
 org.mobyproject.buildkit.worker.hostname:         7b6bf1639e42
 org.mobyproject.buildkit.worker.network:          host
 org.mobyproject.buildkit.worker.oci.process-mode: sandbox
 org.mobyproject.buildkit.worker.selinux.enabled:  false
 org.mobyproject.buildkit.worker.snapshotter:      overlayfs
GC Policy rule#0:
 All:           false
 Filters:       type==source.local,type==exec.cachemount,type==source.git.checkout
 Keep Duration: 48h0m0s
 Keep Bytes:    488.3MiB
...
```

3. Build & Push an image for AMD64 & AMR64

```
$ docker buildx build --push \
  --platform linux/arm64/v8,linux/amd64
\ -t scraly/gophers-api:multi-arch \
  .
```

4. Display the manifest

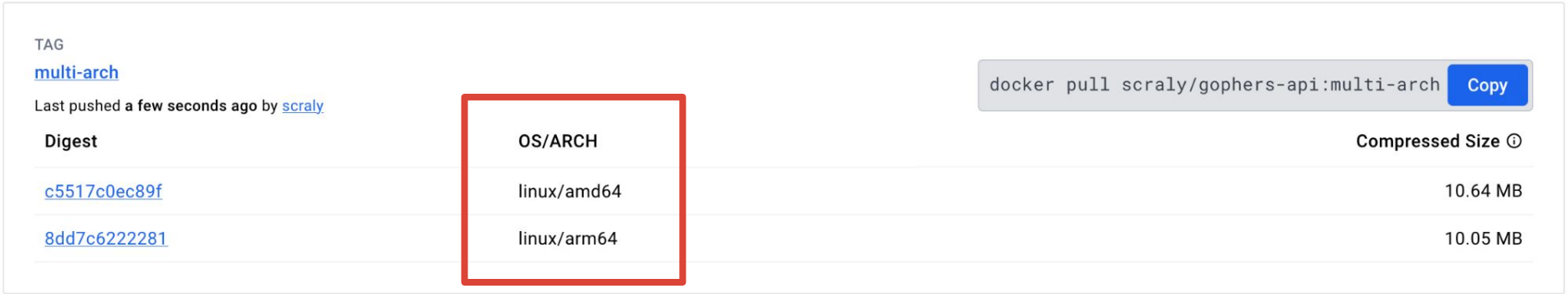
```
$ docker manifest inspect scraly/gophers-api:multi-arch
{
   "schemaVersion": 2,
   "mediaType": "application/vnd.oci.image.index.v1+json",
   "manifests": [
      {
         "mediaType": "application/vnd.oci.image.manifest.v1+json",
         "size": 673,
         "digest":
"sha256:8d[illegible]1fe756fb31e6ffd47832810edd0cdb8f10a63dc23877613c13",
            "architecture": "arm64",
            "os": "linux"
         }
      },
      {
         "mediaType": "application/vnd.oci.image.manifest.v1+json",
         "size": 673,
         "digest":
"sha256:c5[illegible]c60580ed92386f04a7e5636f2f3eaf4594709687bf841c4701",
            "architecture": "amd64",
            "os": "linux"
         }
      },
      ...
   ]
}
```

5. Local verification

```
ache@air-de-aurelie
$ docker pull scraly/gophers-api:multi-arch
multi-arch: Pulling from scraly/gophers-api
c06715fa14fe: Download complete
Digest: sha256:443802f1c64d59df5cc93f655daa90fd75408b164b9366d3d8032509d2147a4e
Status: Downloaded newer image for scraly/gophers-api:multi-arch
docker.io/scraly/gophers-api:multi-arch

What's next:
    View a summary of image vulnerabilities and recommendations → docker scout quickview scraly/gophers-api:multi-arch
```

```
18:22:07 ...managed-private-registry/create-registry-with-pulumi/ovhcloud-tf-registry-go
@air-de-aurelie
$ docker image ls scraly/gophers-api:multi-arch
REPOSITORY           TAG          IMAGE ID       CREATED          SIZE
scraly/gophers-api   multi-arch   443802f1c64d   36 minutes ago   34.3MB
```

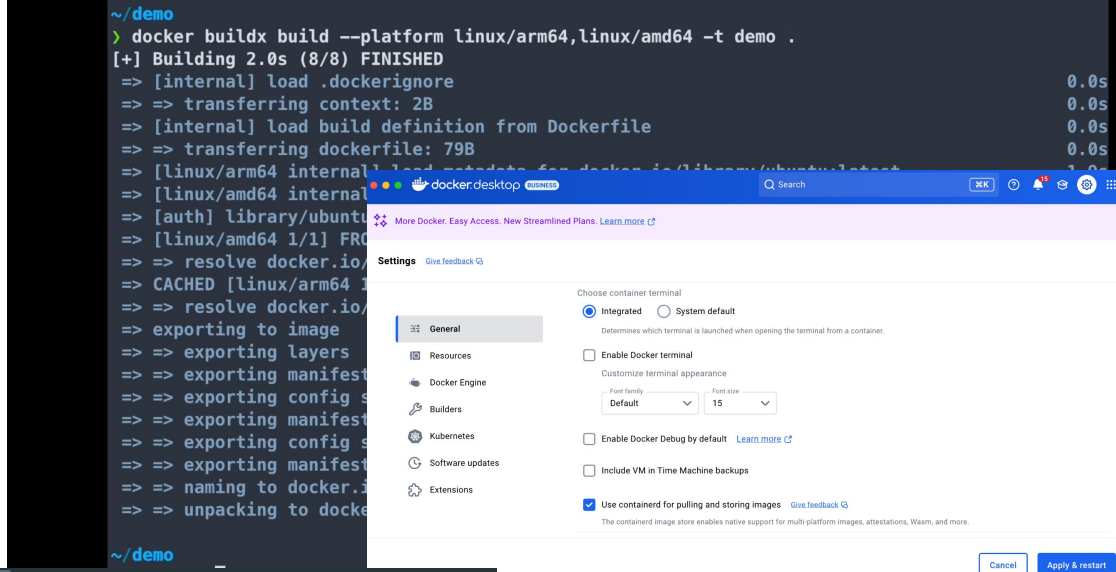
5. Registry verification

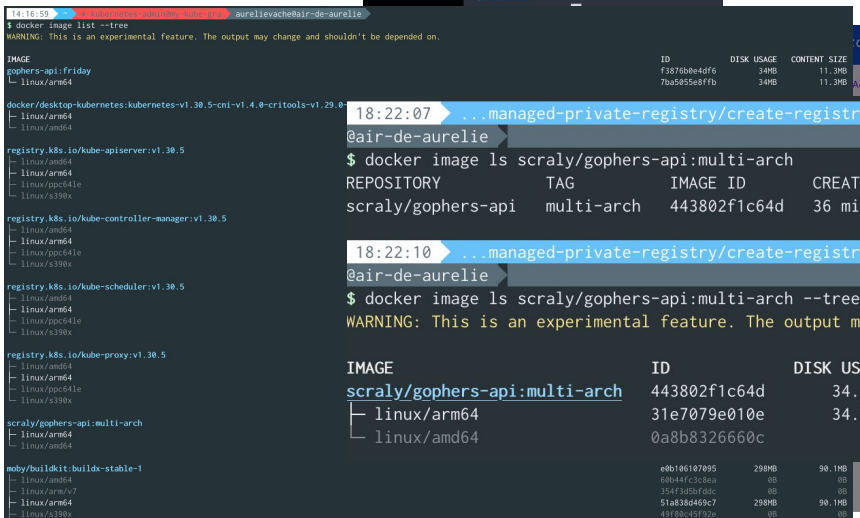


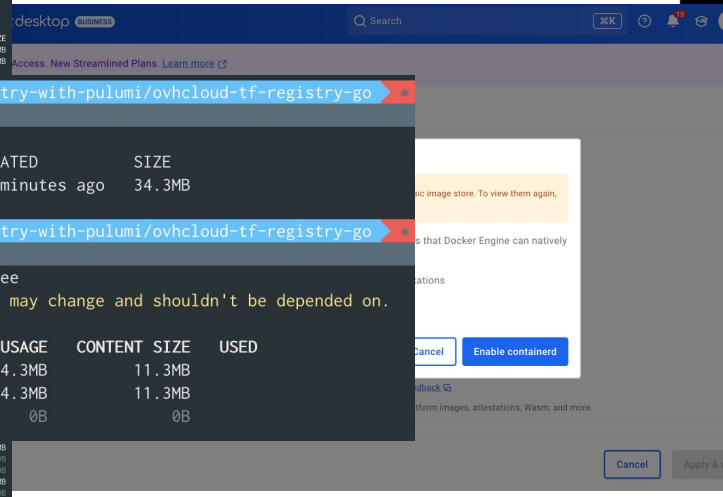
https://hub.docker.com/r/scraly/gophers-api/tags

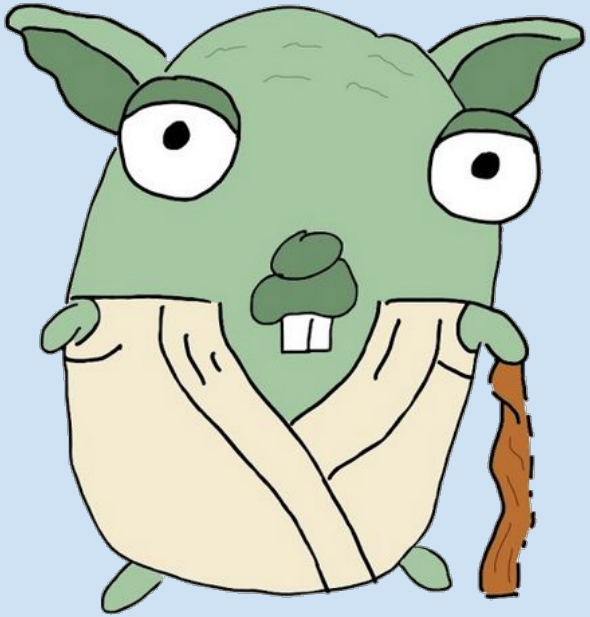
Also (+ easier) we could use **containerd** *experimental feature* as a container store to push / pull

docker image ls --tree helps visualizing multi-arch images in a user friendly way (shows which architecture is used locally)









Question n°7

slido

What are Open Source formats for Software Bill Of Materials (SBOM)?

ⓘ Click **Present with Slido** or install our Chrome extension to activate this poll while presenting.

What are Open Source formats for Software Bill Of Materials (SBOM) ?

A- SPDX

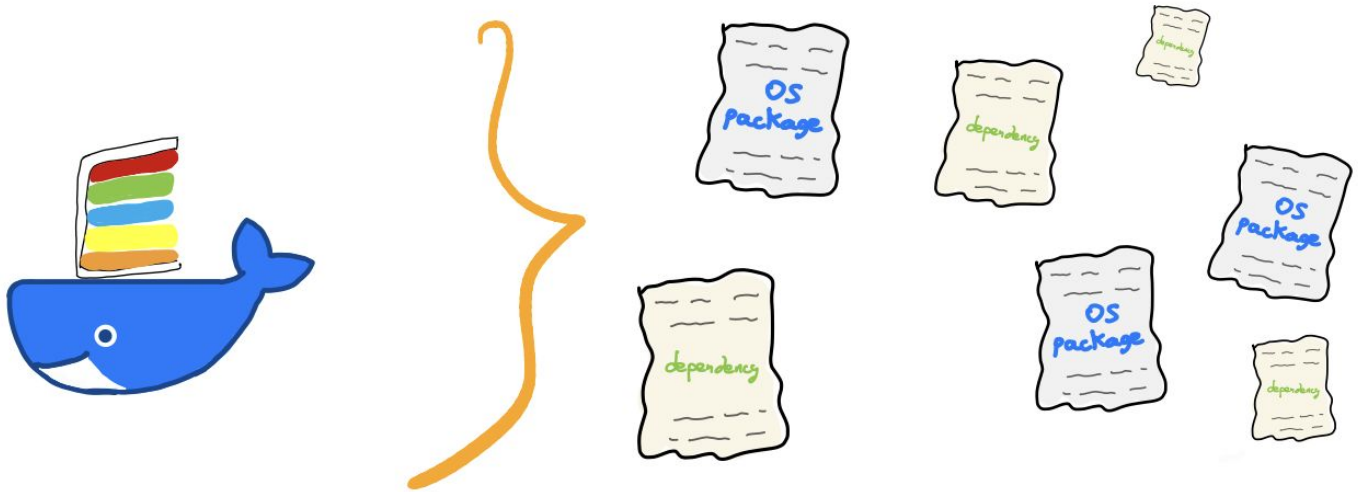
B- Cyclone DX

C- Typhoon DX

D- Syft

SBOM: Software Bill Of Materials

An **inventory** of all components & software dependencies in a container image.



SPDX or CycloneDX ?

SPDX

- Since 2011
- By Linux Foundation
- Used in other domains (Health, Automotive)
- More verbose (licenses, interdependencies, snippets, annotations)
- ISO/IEC 5962 since 2021

CycloneDX

- Since 2017
- By OWASP
- Focused on vulnerability scan / automation
- Less verbose (dependencies, services, components)

🤔 What to choose?

Both are supported by most tools

& OCI doesn't choose for you…

bom

Kubernetes

SPDX

ko

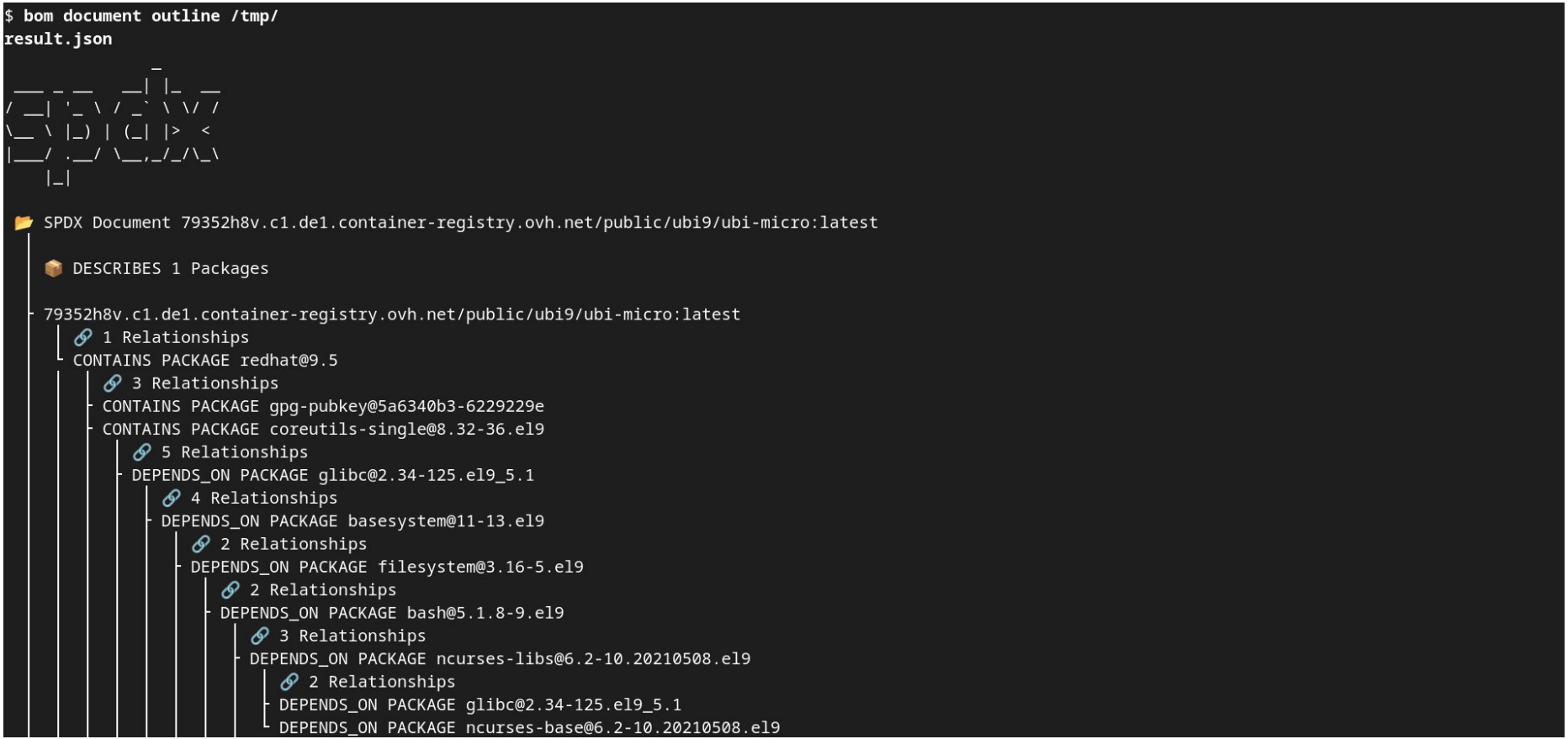
license management

CycloneDX

vulnerabilities

Demo!

```
Login Succeeded!
WARNING! Using --password via the CLI is insecure. Use --password-stdin.

Login Succeeded
$ Générer un SBOM avec Trivy
$ skopeo copy docker://registry.redhat.io/ubi9/ubi-micro:latest docker://79352h8v.c1.de1.container-registry.ovh.n
et/public/ubi9/ubi-micro:latest --remove-signatures

Copying blob 52bc84e86e2d skipped: already exists
Copying blob 700936c8aa4f skipped: already exists
Copying config 4f3b35b7df done   |
Writing manifest to image destination
$ trivy image --format spdx-json --output /tmp/result.json 79352h8v.c1.de1.container-registry.ovh.net/public/ubi9/ubi-micro:latest
2025-01-09T16:09:29+01:00       INFO    "--format spdx-json" disables security scanning. Specify "--scanners vuln" explicitly if you want to include vulnerabi
lities in the "spdx-json" report.
2025-01-09T16:09:30+01:00       INFO    Detected OS     family="redhat" version="9.5"
2025-01-09T16:09:30+01:00       INFO    Number of language-specific files       num=0
```


$ bom document outline /tmp/
result.json
SPDX Document 79352h8v.c1.de1.container-registry.ovh.net/public/ubi9/ubi-micro:latest
DESCRIBES 1 Packages
79352h8v.c1.de1.container-registry.ovh.net/public/ubi9/ubi-micro:latest
1 Relationships
CONTAINS PACKAGE redhat@9.5
3 Relationships
CONTAINS PACKAGE gpg-pubkey@5a6340b3-6229229e
CONTAINS PACKAGE coreutils-single@8.32-36.el9
5 Relationships
DEPENDS_ON PACKAGE glibc@2.34-125.el9_5.1
4 Relationships
DEPENDS_ON PACKAGE basesystem@11-13.el9
2 Relationships
DEPENDS_ON PACKAGE filesystem@3.16-5.el9
2 Relationships
DEPENDS_ON PACKAGE bash@5.1.8-9.el9
3 Relationships
DEPENDS_ON PACKAGE ncurses-libs@6.2-10.20210508.el9
2 Relationships
DEPENDS_ON PACKAGE glibc@2.34-125.el9_5.1
DEPENDS_ON PACKAGE ncurses-base@6.2-10.20210508.el9

```
$ # Tips: on peut utiliser bom, docker et podman pour générer un SBOM
$ # docker scout sbom ubuntu --output sbom.txt
$ # podman build . -t myimage:mytag --sbom-scanner-command trivy --sbom-output sbom.txt
$ # bom generate --image quay.io/skhoury/ubi9/ubi-micro:latest --output=sbom.json --format json
$ bom document outline scout_sbom.json
                _
 ___ _ __   __| |_  __
/ __| '_ \ / _` \ \/ /
\__ \ |_) | (_| |>  <
|___/ .__/ \__,_/_/\_\
    |_|

 📂 SPDX Document quay.io/skhoury/ubi9/ubi-micro

   📦 DESCRIBES 1 Packages

   ├ quay.io/skhoury/ubi9/ubi-micro@latest
   │  🔗 27 Relationships
   │  ├ CONTAINS PACKAGE acl@2.3.1-4.el9
   │  │  🔗 2 Relationships
   │  │  ├ CONTAINS FILE usr/lib64/libacl.so.1.1.2301 (usr/lib64/libacl.so.1.1.2301)
   │  │  └ OTHER FILE var/lib/rpm/rpmdb.sqlite (var/lib/rpm/rpmdb.sqlite)
   │  │
   │  ├ CONTAINS PACKAGE attr@2.5.1-3.el9
   │  │  🔗 3 Relationships
   │  │  ├ CONTAINS FILE etc/xattr.conf (etc/xattr.conf)
   │  │  ├ CONTAINS FILE usr/lib64/libattr.so.1.1.2501 (usr/lib64/libattr.so.1.1.2501)
   │  │  └ OTHER FILE var/lib/rpm/rpmdb.sqlite (var/lib/rpm/rpmdb.sqlite)
   │  │
   │  ├ CONTAINS PACKAGE basesystem@11-13.el9
   │  │  🔗 1 Relationships
   │  │  └ OTHER FILE var/lib/rpm/rpmdb.sqlite (var/lib/rpm/rpmdb.sqlite)
```

```
$ bom document outline bom_sbom.json
                _
 ___ _ __   __| |_  __
/ __| '_ \ / _` \ \/ /
\__ \ |_) | (_| |>  <
|___/ .__/ \__,_/_/\_\
    |_|

 📂 SPDX Document SBOM-SPDX-51cd2088-9f76-4894-8d72-66c6ddfdee5a

    📦 DESCRIBES 1 Packages

   ├ sha256:3313e52bb1aad4017a0c35f9f2ae35cf8526eeeb83f6ecbec449ba9c5cb9cb07
     │ 🔗 1 Relationships
     └ CONTAINS PACKAGE sha256:30332f5989896482be3190c23a6b4cc76e3630b4abfb7e3e1cdc361ceb500a26
       │ │ 🔗 20 Relationships
       ├ CONTAINS PACKAGE redhat-release@9.5-0.6.el9
       ├ CONTAINS PACKAGE setup@2.13.7-10.el9
       ├ CONTAINS PACKAGE filesystem@3.16-5.el9
       ├ CONTAINS PACKAGE libgcc@11.5.0-2.el9
       ├ CONTAINS PACKAGE basesystem@11-13.el9
       ├ CONTAINS PACKAGE tzdata@2024b-2.el9
       ├ CONTAINS PACKAGE pcre2-syntax@10.40-6.el9
       ├ CONTAINS PACKAGE ncurses-base@6.2-10.20210508.el9
       ├ CONTAINS PACKAGE glibc-minimal-langpack@2.34-125.el9_5.1
       ├ CONTAINS PACKAGE glibc-common@2.34-125.el9_5.1
       ├ CONTAINS PACKAGE glibc@2.34-125.el9_5.1
       ├ CONTAINS PACKAGE ncurses-libs@6.2-10.20210508.el9
       ├ CONTAINS PACKAGE bash@5.1.8-9.el9
       ├ CONTAINS PACKAGE libattr@2.5.1-3.el9
       ├ CONTAINS PACKAGE libacl@2.3.1-4.el9
       ├ CONTAINS PACKAGE libcap@2.48-9.el9_2
```

```
$ # Attachons le sbom à l\image
$ oras attach 79352h8v.c1.de1.container-registry.ovh.net/public/ubi9/ubi-micro:latest --artifact-type application/spdx+json trivy_sbom.json:text/spdx+json
Exists    d238c35e5894 trivy_sbom.json
Error: GET "https://79352h8v.c1.de1.container-registry.ovh.net/v2/public/ubi9/ubi-micro/referrers/sha256:0000000000000000000000000000000000000000000000000000000000000000": response status code 401: unauthorized: un-recognized request: GET /v2/public/ubi9/ubi-micro/referrers/sha256:0000000000000000000000000000000000000000000000000000000000000000: un-recognized request: GET /v2/public/ubi9/ubi-micro/referrers/sha256:0000000000000000000000000000000000000000000000000000000000000000
$ # Regardons la registry... https://79352h8v.c1.de1.container-registry.ovh.net/harbor/projects/2/repositories/ubi9%2Fubi-micro/artifacts-tab
$ oras discover 79352h8v.c1.de1.container-registry.ovh.net/public/ubi9/ubi-micro:latest
Error: GET "https://79352h8v.c1.de1.container-registry.ovh.net/v2/public/ubi9/ubi-micro/referrers/sha256:03692dc74d7a1870540bfe92ebded314cfd37b72d6174935c9aaf65d6b875269": response status code 401: unauthorized: un-recognized request: GET /v2/public/ubi9/ubi-micro/referrers/sha256:03692dc74d7a1870540bfe92ebded314cfd37b72d6174935c9aaf65d6b875269: un-recognized request: GET /v2/public/ubi9/ubi-micro/referrers/sha256:03692dc74d7a1870540bfe92ebded314cfd37b72d6174935c9aaf65d6b875269
$ oras discover sherinefedora:5000/ubi9/ubi-micro:latest
Discovered 1 artifact referencing latest
Digest: sha256:3313e52bb1aad4017a0c35f9f2ae35cf8526eeeb83f6ecbec449ba9c5cb9cb07

Artifact Type                       Digest
application/spdx+json   sha256:0210b5f0d75da9cd5eda61002c0bf5872ac588a7629d87e18635f8bd8d1df4a0$ Fini !
```

LAST QUESTION

Question n°8

slido

With cosign, when I sign an image, the signature is:

ⓘ Click **Present with Slido** or install our Chrome extension to activate this poll while presenting.

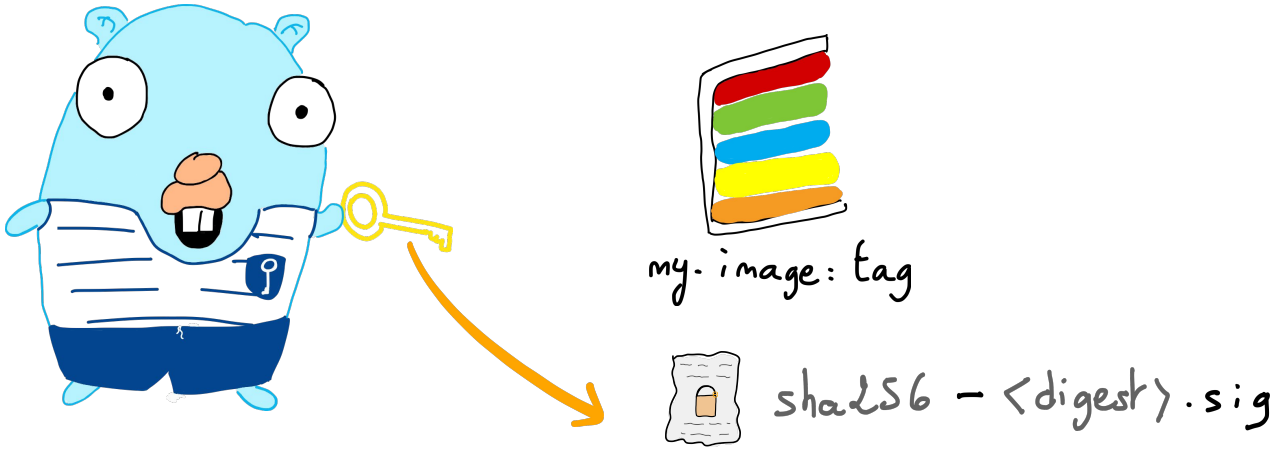
With cosign, when I sign an image, the signature is:

A- An extra layer added in the image
B- A new image tag
C- A copy of the image

Sign An Image (without modifying it) with Cosign

"Sign OCI containers"

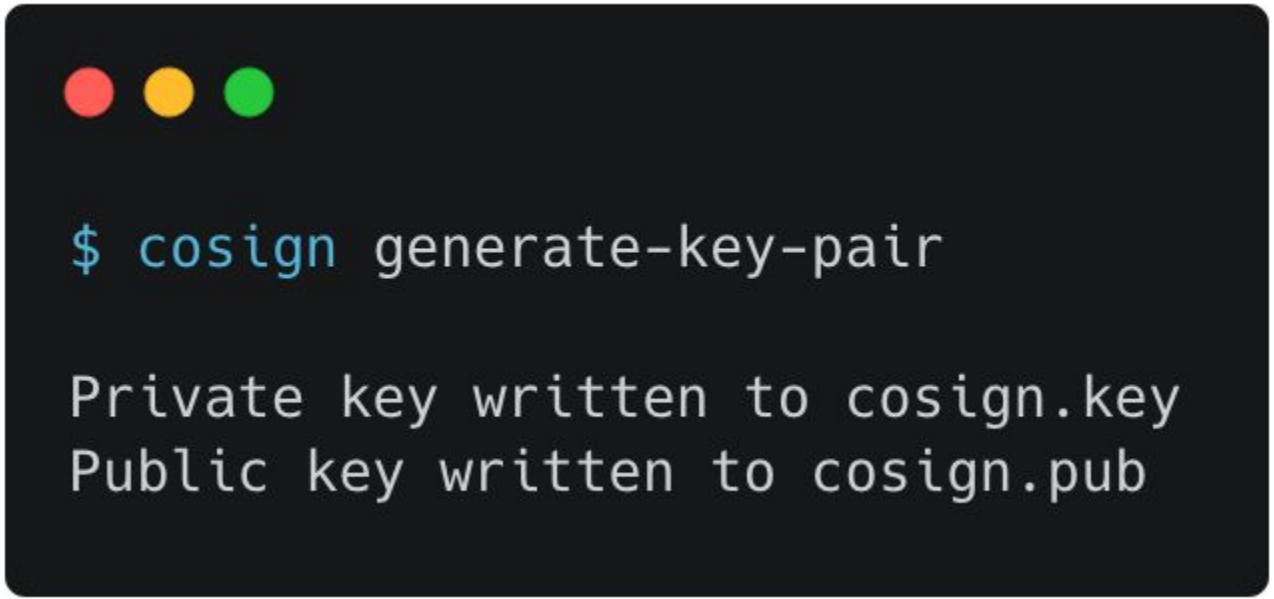
Cosign (Sigstore) handles signatures for OCI container images.



A signature is a new image tag, with its metadata stored on a separate server (Rekor)

Demo!

1. Generate a key pair



2. Get the image digest

```
$ docker inspect 79352h8v.c1.de1.container-registry.ovh.net/public/gophers-api
| jq -r '.[0].RepoDigests[0]'

79352h8v.c1.de1.container-registry.ovh.net/public/gophers-
api@sha256:b229a8c3a6b9fc5873c97e60636e58fa0cee8cec449104cf3fa2bfa6639f3833
```

```
$ IMG_DIGEST=$(docker inspect 79352h8v.c1.de1.container-registry.ovh.net/public/gophers-api | jq -r '.[0].RepoDigests[0]')
```

3. Sign the OCI image

```
# sign the OCI artifact and push to the Managed Private Registry/Harbor instance
# and store the transparency log (metadata) in the public Rekor server
# at https://rekor.sigstore.dev/ (to verify the signature afterward)
$ cosign sign --key cosign.key 79352h8v.c1.de1.container-registry.ovh.net/public/gophers-
api@sha256:b229a8c3a6b9fc5873c97e60636e58fa0cee8cec449104cf3fa2bfa6639f3833

tlog entry created with index: 158124855
Pushing signature to: 79352h8v.c1.de1.container-registry.ovh.net/public/gophers-api
```

4. Verify the image

Via the CLI :

```
$ cosign verify 79352h8v.c1.de1.container-registry.ovh.net/public/gophers-api --key cosign.pub
-o text | jq

{
  "critical": {
    "identity": {
      "docker-reference": "79352h8v.c1.de1.container-registry.ovh.net/public/gophers-api"
    },
    "image": {
      "docker-manifest-digest":
"sha256:b229a8c3a6b9fc5873c97e60636e58fa0cee8cec449104cf3fa2bfa6639f3833"
    },
    "type": "cosign container image signature"
  },
  "optional": null
}
```

4. Verify the image

On OVHcloud Managed Private Registry (Harbor) :

Artifacts	Pull Command	Tags	Signed by Cosign	Size	Vulnerabilities	Annotations	Labels
sha256:893b5a7c				10.06MiB	Not Scanned		
sha256:b229a8c3		latest		10.85MiB	Not Scanned		

< Projects < public < gophers-api

sha256:7256ab80

Tags

+ ADD TAG | REMOVE TAG

Name	Pull Time	Push Time
sha256-74a2d788b859dfd74007eb9665f3645d6ff4768e721ca3653c0b00eeb75d0204.sig	28/01/2025 18:23	28/01/2025 18:23

Page size 15 1 - 1 of 1 items

5. Display the signature tag : Format sha256-<digest>.sig

```
$ cosign triangulate 79352h8v.c1.de1.container-registry.ovh.net/public/gophers-api

79352h8v.c1.de1.container-registry.ovh.net/public/gophers-api:sha256-
b229a8c3a6b9fc5873c97e60636e58fa0cee8cec449104cf3fa2bfa6639f3833.sig
```

6. Inspect the manifest of this special tag

```
$ crane manifest 79352h8v.c1.de1.container-registry.ovh.net/public/gophers-
api:sha256-b229a8c3a6b9fc5873c97e60636e58fa0cee8cec449104cf3fa2bfa6639f3833.sig |
jq .

{
  "schemaVersion": 2,
  "mediaType": "application/vnd.oci.image.manifest.v1+json",
  "config": {
    "mediaType": "application/vnd.oci.image.config.v1+json",
    "size": 560,
    "digest":
"sha256:7b560855289adf2a5988c7c4fc207c54c7e899517974cb294422731c7884fead"
  },
  "layers": [
    {
      "mediaType": "application/vnd.dev.cosign.simplesigning.v1+json",
      "size": 292,
      "digest":
"sha256:470dcd201aa0842fe8a16b898dee0010cbfb8e8f78f285825360da8d058eef9f",
      "annotations": {
        "dev.cosignproject.cosign/signature": "MEUCIQCeG6GwmbkI+OhC/
lKUYmvCvr7wcVaQnzDGa72yV1N5OwIgeLx1Szpjx1ZRP9BIqfTbemqPzOhQruZnLz832oqM0rQ=",
        "dev.sigstore.cosign/bundle": "{\"SignedEntryTimestamp\":
\"MEUCIQDmdyV8c8AU1fnBZTg4hg2SBbUYx0i5EN+XswID6K11ywIgbR8cPUcqkWcTVuK43MmKDcI0YAhkB
Bz61MRouI9X+s0=\",\"Payload\":{\"body\":\"<body>\",\"integratedTime\":1733844329,
\"logIndex\":154484812,\"logID\":
\"c0d23d6ad406973f9559f3ba2d1ca01f84147d8ffc5b8445c224f98b9591801d\"}}"
      }
    }
  ]
}
```

7. Tips: You can even add an annotation/information to the signature

```
$ cosign sign -y -a conf=tnt --key cosign.key $IMG_DIGEST
```

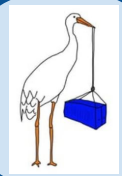
7. Tips: On peut meme ajouter une annotation/information a notre signature

```
$ cosign verify 79352h8v.c1.de1.container-registry.ovh.net/public/gophers-api
--key cosign.pub -o text | jq
```


```
{
  "critical": {
    "identity": {
      "docker-reference": "79352h8v.c1.de1.container-registry.ovh.net/public/
gophers-api"
    },
    "image": {
      "docker-manifest-digest":
"sha256:b229a8c3a6b9fc5873c97e60636e58fa0cee8cec449104cf3fa2bfa6639f3833"
    },
    "type": "cosign container image signature"
  },
  "optional": null
}
{
  "critical": {
    "identity": {
      "docker-reference": "79352h8v.c1.de1.container-registry.ovh.net/public/
gophers-api"
    },
    "image": {
      "docker-manifest-digest":
"sha256:b229a8c3a6b9fc5873c97e60636e58fa0cee8cec449104cf3fa2bfa6639f3833"
    },
    "type": "cosign container image signature"
  },
  "optional": {
    "conf": "snowcamp"
  }
}
```

BONUS TRACK

Each Tool has a super power



Crane

> Flatten images
> Add layers without Dockerfile

Trivy

> Scan vulnerabilities
> Generate a SBOM

Skopeo

> Copy images, from a registry to another

Cosign

> Sign images, even with your GitHub account

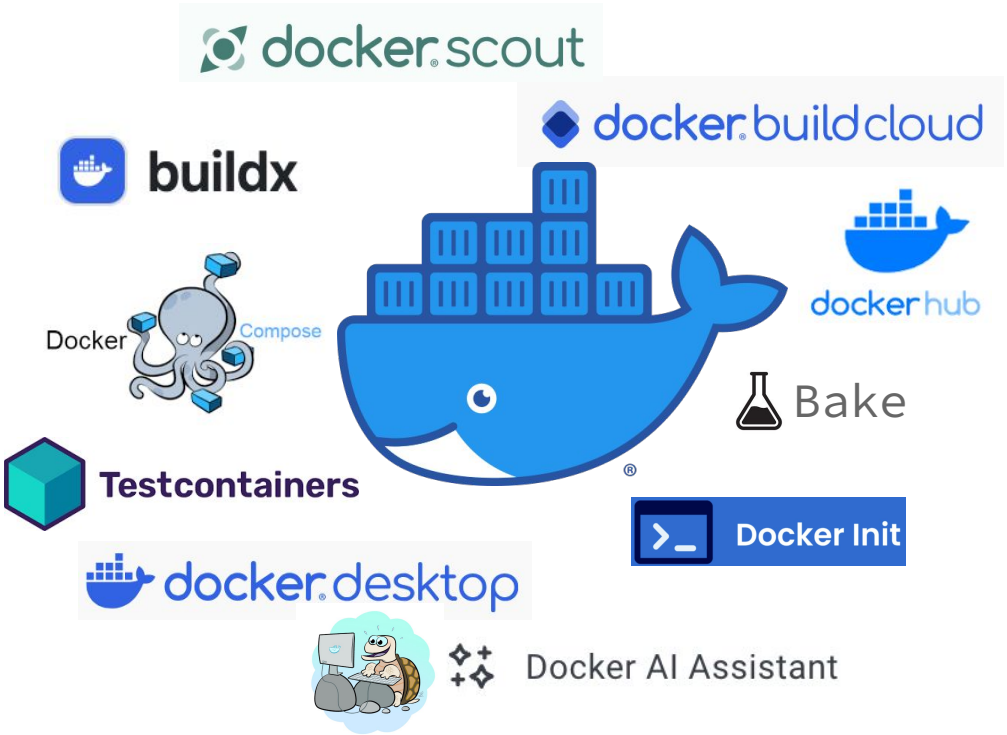
BOM

> Read/generate a SBOM

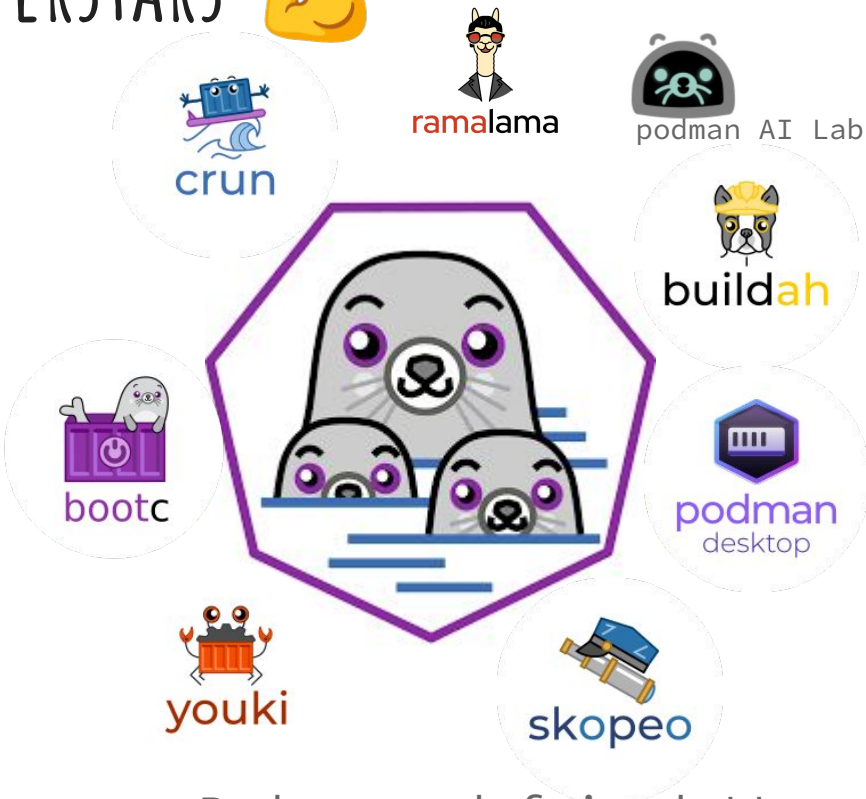
ORAS

> Bake any artifact into an image

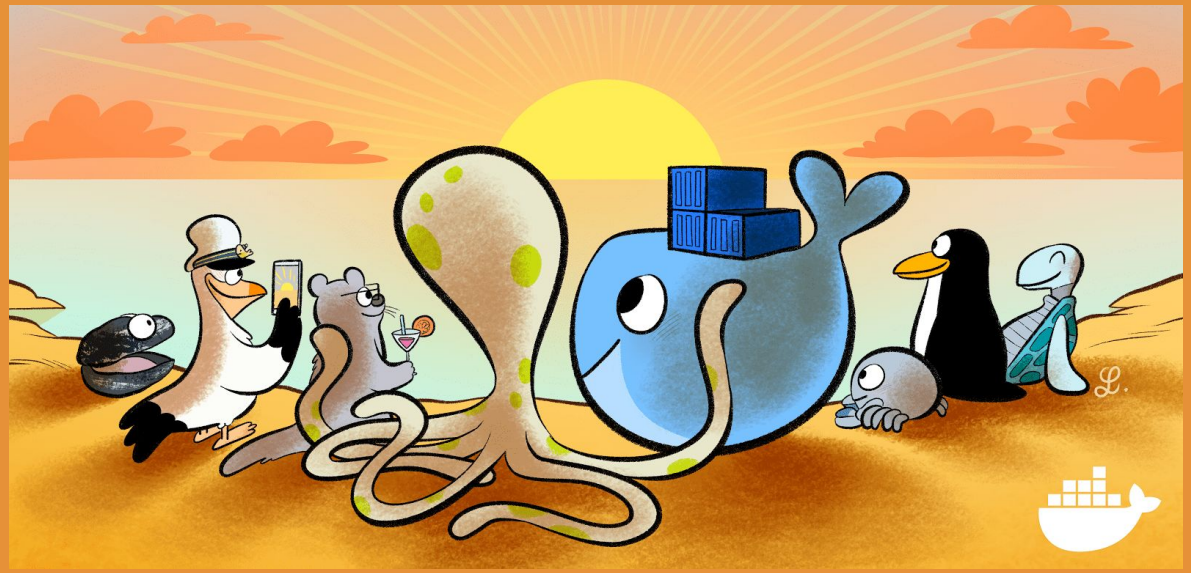
Last but not least, our two superstars 💪



Mature
THE reference



Podman and friends!!
No d(a)emons 😈



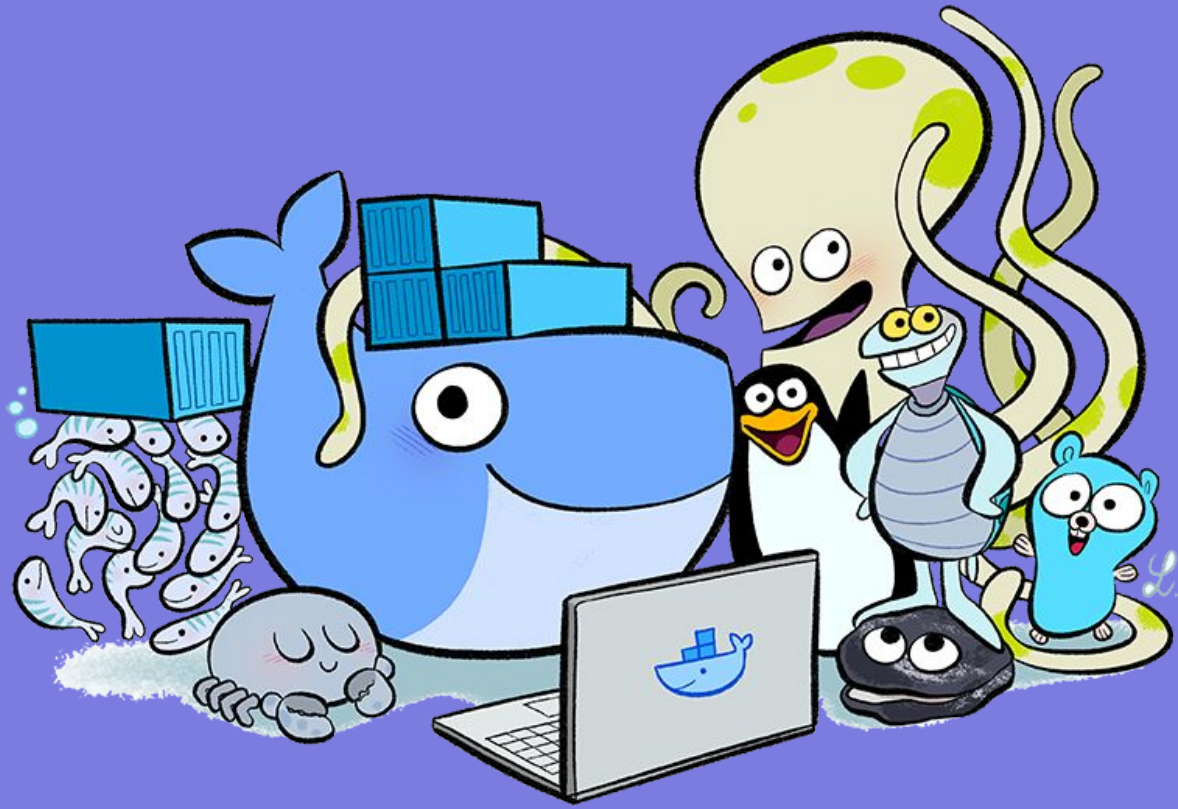
Conclusion

$ DOCKER BUILD "THANKSSS"

https://ovh.to/tFHhyo

Please scan the QR Code to leave feedback on this session ❤️

@aurelievache | @srinerine



$ DOCKER RUN QUESTIONS