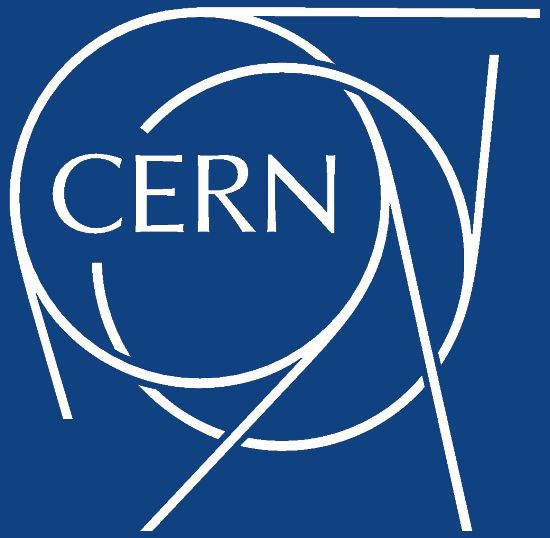

CERN

CVMFS remote snapshotter

Spyridon Trigazis
Theodoros Tsioutsias

Outline

1. Motivation
2. CVMFS Prometheus metrics
3. K8S configuration
4. Remote snapshotter interface
5. CVMFS repo layout
6. Benchmark
7. Next Steps

Motivation

1. ~76% of container startup time is the pull of the image [FAST '16]
2. Most of the times not everything included in an image is needed
3. Properly test the integration of remote snapshotters with K8S
4. We wanted to explore all the available options and compare them:
   a. CVMFS backed registry
   b. CVMFS - CRFS backed registry
   c. Stock containerd

CVMFS Prometheus metrics

We implemented a prometheus exporter for CVMFS publishing stats:

- cvmfs_cached_bytes
- cvmfs_pinned_bytes
- cvmfs_expires
- cvmfs_inode_max
- cvmfs_maxfd
- cvmfs_ncleanup24
- cvmfs_nclg
- cvmfs_ndiropen
- cvmfs_ndownload
- cvmfs_nioerr
- cvmfs_nopen
- cvmfs_pid
- cvmfs_revision
- cvmfs_rx
- cvmfs_speed
- cvmfs_timeout
- cvmfs_timeout_direct
- cvmfs_uptime
- cvmfs_usedfd

Statistics are based on cvmfs_talk and the extended CVMFS attributes.

The producer will be available for users on K8S.

K8S configuration

- Containerd already includes a CRI plugin
  - https://github.com/containerd/cri/blob/master/docs/installation.md
- Uses different API e.g. crictl to interact with containerd
- Hacked containerd to make it preselect remote snapshotter
  - make it a proper configuration option
- Deployed CVMFS as a daemonset
  - Expose the metrics with prometheus service monitor
- Build containerd against fedora's libseccomp, glibc-static
  - https://gitlab.cern.ch/cloud/remote-snapshotter/tree/cern

```
NAME                                STATUS VERSION   CONTAINER-RUNTIME
compute-opt-stargz-03-minion-0      Ready  v1.15.3   containerd://1.3.0-132-gf6992afb
compute-opt-stock-03-minion-0       Ready  v1.15.3   containerd://1.3.0-132-gf6992afb
compute-opt-unpacked-03-minion-0 Ready  v1.15.3   containerd://1.3.0-132-gf6992afb
```

Remote snapshotter interface

- Containerd plugin
- Currently a shared library (.so) loaded at containerd startup
- Ability to move this functionality to gRPC
- Defines an interface for mounting remote snapshots

CVMFS snapshotter:

- All layers are unpacked in CVMFS
- Looks for the snapshot under CVMFS mountpoint

CVMFS - CRFS snapshotter:

- Compressed data in stargz format
- Stargz:
  - means seekable .tar.gz
  - is a tarball containing taballs (tar of tars)
  - originally implemented for CRFS

CVMFS repo layout

```
unpacked/
├── 0a
│   └── 0a27cd0a6fb4b4e1ea7c6369d6e8058b9212bd21a1a1a755345549863f8f6df0
│       ├── opt
│       ├── sw
│       ├── tmp
│       ├── usr
│       └── var
...

stargz/
└── blobs
    └── sha256
        ├── 03
        │   └── 03407a3d83f2350bd9e4e23c0f139802b8231548f6c593999e6208f8a6e6f6a0
        │       └── data
...
```

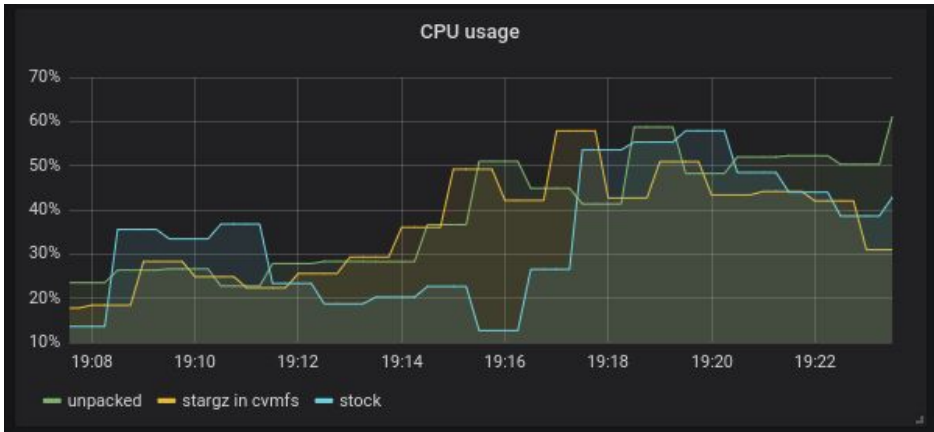
Benchmark

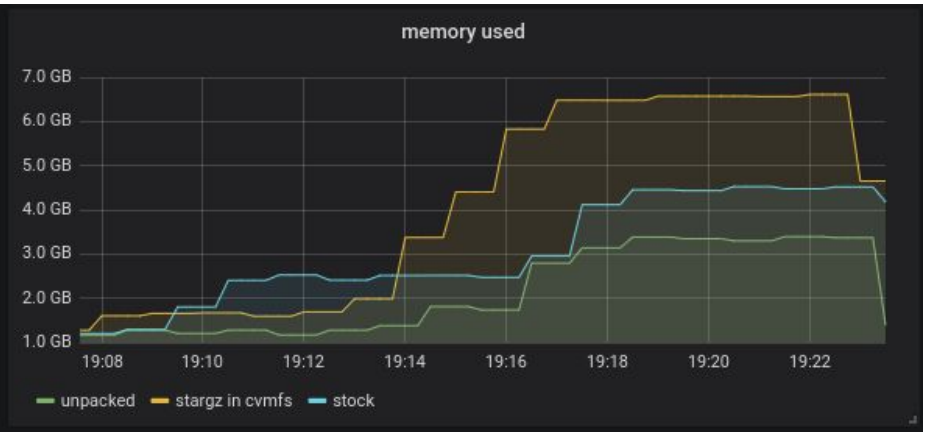
Overall performance measured base on:

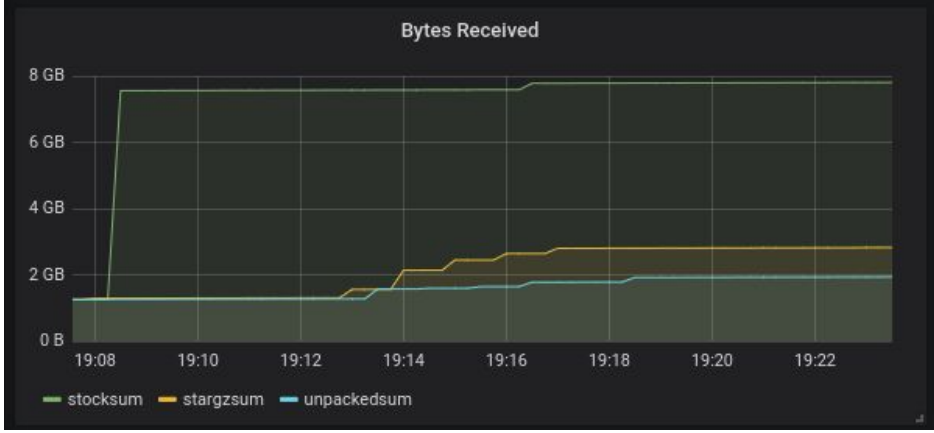
- CPU usage
- RAM usage
- CVMFS cache
- Network traffic
- atlas/athena:21.0.15_100.0.2

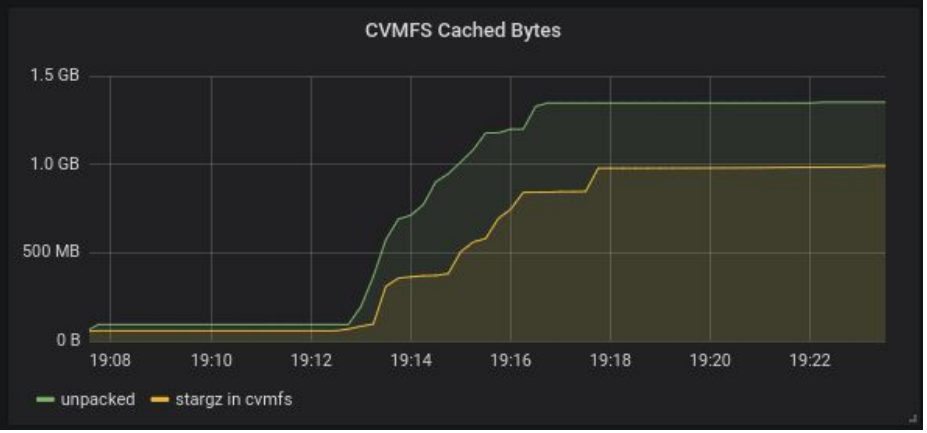
cond mode	cvmfs_cache	cvmfs storage	container startup	CPU usage	RAM usage	bytes received	job runtime
cvmfs	980MB	17GB	1s	36%	2.0GB	1.936GB	10m22s
crfs-cvmfs	1.347GB	5.5GB	5s	37%	3.9GB	2.824GB	09m14s
stock	-	-	03m15s	34%	2.9GB	7.807GB	07m34s

Benchmark


CPU usage
70%
60%
50%
40%
30%
20%
10%
19:08
19:10
19:12
19:14
19:16
19:18
19:20
19:22
unpacked
stargz in cvmfs
stock


memory used
7.0 GB
6.0 GB
5.0 GB
4.0 GB
3.0 GB
2.0 GB
1.0 GB
19:08
19:10
19:12
19:14
19:16
19:18
19:20
19:22
unpacked
stargz in cvmfs
stock

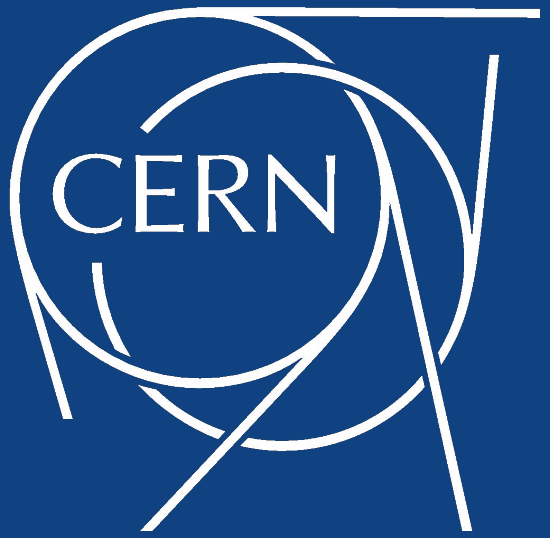

Bytes Received
8 GB
6 GB
4 GB
2 GB
0 B
19:08
19:10
19:12
19:14
19:16
19:18
19:20
19:22
stocksum
stargzsum
unpackedsum


CVMFS Cached Bytes
1.5 GB
1.0 GB
500 MB
0 B
19:08
19:10
19:12
19:14
19:16
19:18
19:20
19:22
unpacked
stargz in cvmfs

Next Steps

- Remote snapshotters are still a work in progress
- Cooperate with upstream to push this forward
- Stress test containerd for stability
  - make containerd default CRI (vs current docker)
- Use base image from CVMFS
  - Additional layers from gitlab-registry
    - Develop on top of atlas/athena:21.0.15_100.0.2
    - Push extra layer(s) to gitlab registry
    - Pull athena from CVMFS and additional layers from gitlab

Thank you!


CERN