DOST

Container, Cloud & Co.

24. – 26. SEPTEMBER BERLIN

Michel Raabe

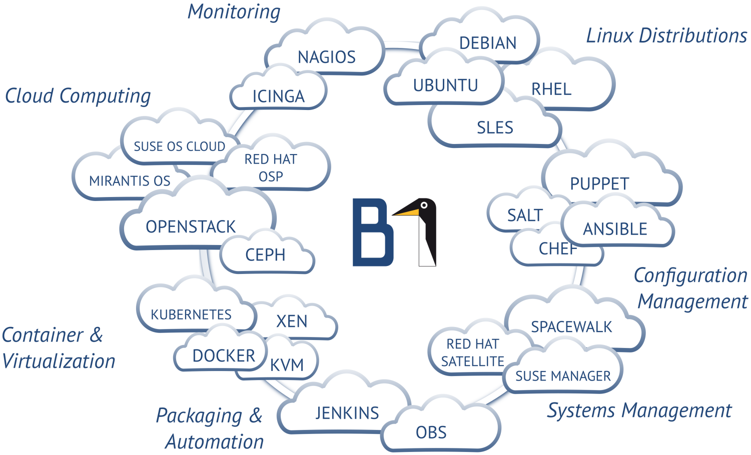
B1 Systems GmbH

Ceph Backups mit Ceph-zu-Ceph

Introducing B1 Systems

- founded in 2004
- operating both nationally and internationally
- more than 100 employees
- vendor-independent (hardware and software)
- focus:
    - consulting
    - support
    - development
    - training
    - operations
    - solutions
- branch offices in Rockolding, Berlin, Cologne & Dresden

Areas of Expertise



Monitoring

NAGIOS

ICINGA

Linux Distributions

DEBIAN

UBUNTU

RHEL

SLES

Cloud Computing

SUSE OS CLOUD

RED HAT OSP

MIRANTIS OS

OPENSTACK

CEPH

Configuration Management

PUPPET

SALT

ANSIBLE

CHEF

Container & Virtualization

KUBERNETES

XEN

DOCKER

KVM

Systems Management

SPACEWALK

RED HAT SATELLITE

SUSE MANAGER

Packaging & Automation

JENKINS

OBS

Running backups with Ceph-to-Ceph

Requirements

- “We want to backup our Ceph cluster”
- independent from OpenStack
- asynchronous
- not always offsite
- fuc**** file browser

Methods

native

- rbd-mirror
- s3 multisite

3rd party

- “scripts”
- backy2
- rbd2qcow
- ???

Challenges

Challenges

- crash consistency
- disaster recovery
- bandwidth
- costs

Crash consistency

- only access to the base layer
- unknown workload
- corrupt filesystem
- lost transactions

Disaster recovery

- how to access the remote cluster?
  - bandwidth?
  - route?
- switch the storage backend?
  - supported by the application?

Bandwidth

- bandwidth vs. backup volume
  - 20TB in 24h over 800mbit - no
- network latency

Costs

- a second cluster
  - different type of disks (hdd/ssd/nvme)
- similar amount of available disk space
- uplink
  - 1/10/100 Gbit

What can we do?

rbd-mirror – Overview

- does mirror support:
  - single image
  - whole pool
- available since jewel (10.2.x)
- asynchronous
- daemon: rbd-mirror
- no “file browser”

rbd-mirror – What's needed?

- rbd feature: journaling (+ exclusive lock):

```
rbd feature enable ... journaling
```

- 30s default trigger:

```
rbd_mirror_sync_point_update_age
```

- cluster name
  - default is "ceph"
  - it's possible but ...
  - ... hard to track

rbd-mirror – Keyring sample

- key layout:

```
ceph.client.admin.keyring
<clustername>.<type>.<id>.keyring
```

- example config files:

```
remote.client.mirror.keyring
remote.conf
local.client.mirror.keyring
local.conf
```

rbd-mirror – Problems

- rbd-daemons must be able to connect to both clusters
- no two public networks
- same network or
- routing kung fu
- krbd module

s3 multisite – Overview

- S3 - simple storage service
- compatible with Amazon S3
- Swift compatible
- Keystone integration
- encryption
- no “file browser”

s3 multisite – What's needed?

- read-write or read-only
- cloudsync plugin (e.g. aws)
- NFS export possible
- S3 "client"

s3 multisite – Problems

- Masterzone “default”
- Zonegroup(s) “default”

```
<zone>-<zonegroup1> <-> <zone>-<zonegroup2>
default-default <-> default-default
de-fra <-> de-muc
```

- Zonegroups are synced
- one connection ...
- ... radosgw to radosgw

3rd party

- custom scripts
- backy2
- ...

all based on snapshots and diff exports

3rd party – Scripts – Overview

- should use snapshot and ‘diff’

```
rbd snap create ..
rbd export-diff .. | ssh rbd import-diff ..
rbd export-diff --from-snap .. | ssh rbd import-diff ..
```

- someone has to track the snapshots

3rd party – backy2 – Overview

- internal db
- snapshots
- rbd and file
- can do backups to s3
- tricky with k8s
- python (only deb)
- nbd mount

3rd party – Problems

- active/old snapshots
- k8s with ceph backend
- pv cannot be deleted
- tracking/database
- still no “file browser”

3rd party – Example workflow

1. Create initial snapshot.
2. Copy first snapshot to remote cluster.
3. ... *next hour/day/week/month ...*
4. Create a snapshot.
5. Copy diff between snap1 vs snap2 to remote cluster.
6. Delete old snapshots.

What can't we do

rbd export

- plain rbd export
- for one-time syncs
- only disk images

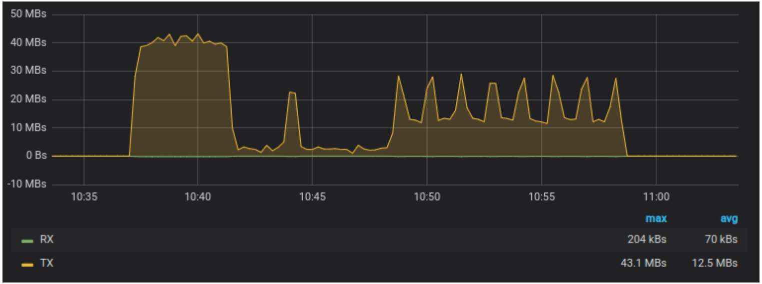


Figure: rbd export | rbd import - 21min (20G)

rbd export-diff

- plain rbd export-diff
- depends on the (total) diff size
- only disk images
- fast(er)?
- scheduled

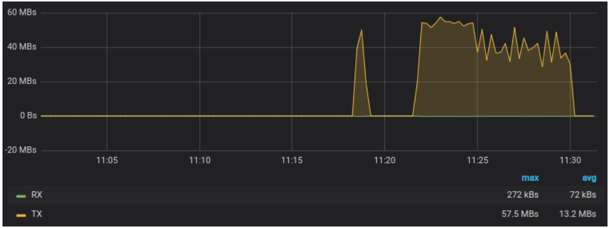


Figure: rbd export-diff - 8min (20G)

rbd mirror

- rbd-mirror
- slow?
- runs in the background
- no schedule

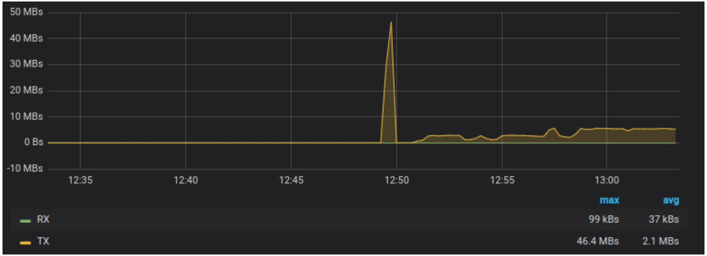


Figure: rbd mirror - 30min (20G)

s3 multisite

- s3 capable client
- “only” http(s)
- scalable (really, really well)

Wrap-up

What now?

1. rbd snapshots
    - simple, easy, fast
    - controllable
    - can be a backup
2. s3 multisite
    - simple, fast
    - built-in
    - nfs export (ganesha)
    - no backup at all
3. rbd-mirror
    - disk-based
    - built-in
    - no backup at all

What's with ...

- cephfs
  - no built-in replication
  - hourly/daily rsync?
  - snapshots?

Thank You!

For more information, refer to info@b1-systems.de
or +49 (0)8457 - 931096