



CERNBox backup evolution

Summary

- Requirements
- cback backup system
- Groups
- Portal
- Future works

Requirements

- **Disaster recovery:** the system should help in case of software/natural disasters
- **Scalability:** the system should scale with the services being backed up
- Ensure **daily backups**
- **Low costs** of the system: everything should be automatized, low load on operators
 - Restore should be as simple as possible
- **Low impact** on services
- **Data retention:** oldest backups should be deleted according to a retention policy
- Support for **multiple storages**
- **Interoperability** with other software

cback backup system

Introduction

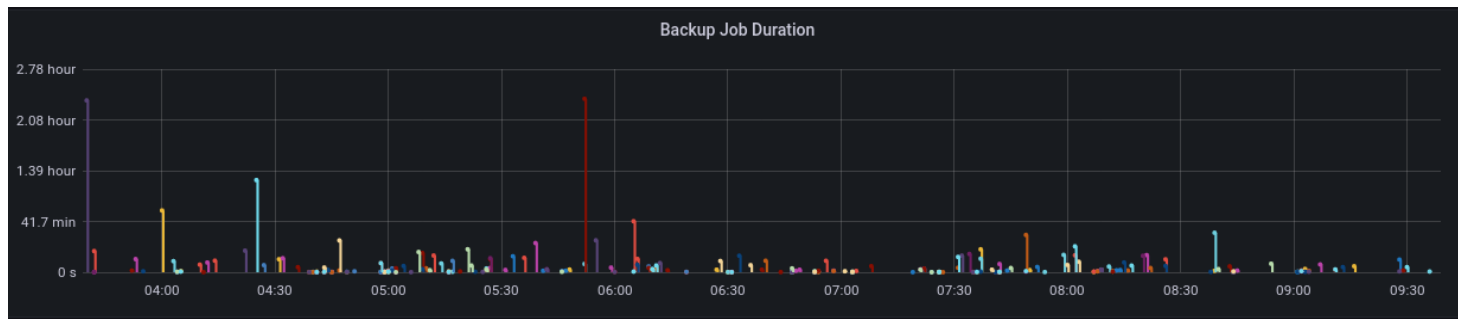
- **Backup orchestration tool** developed at CERN
- Creates **daily snapshots** for all CERNBox users and projects spaces and some buckets in Ceph
- Based on **restic**:
 - Fast
 - Easy
 - Repository/snapshots based
 - Deduplication
 - Encryption
 - Prune old snapshots
 - Support for multiple backends



cback backup system

Overview

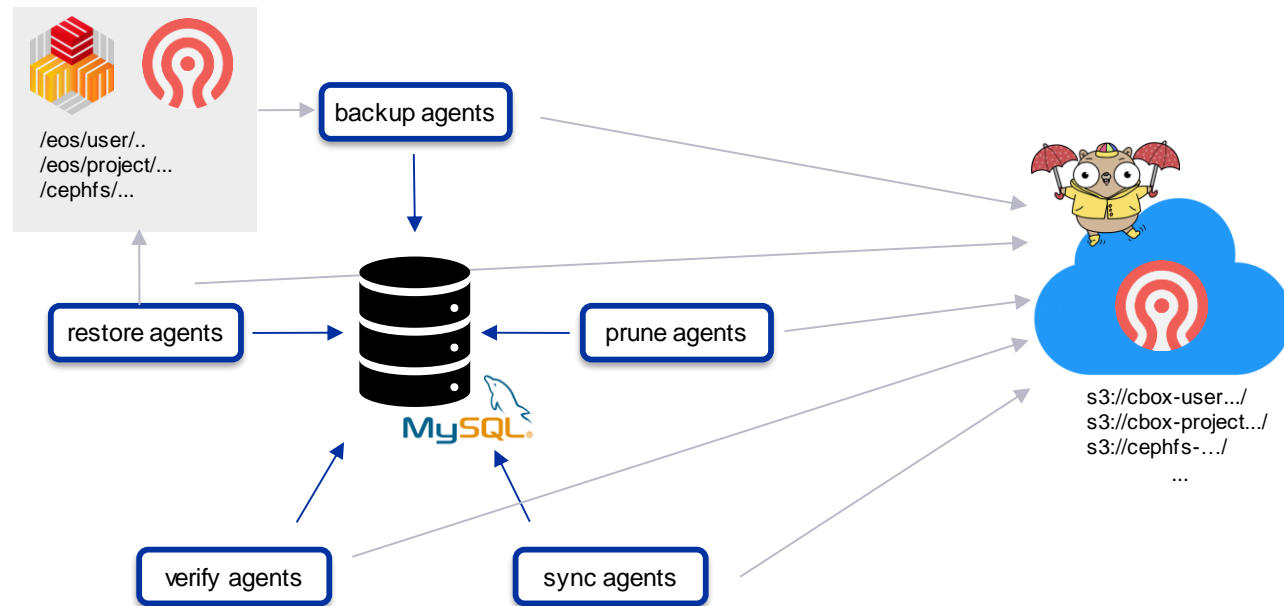
- Data stored in an **other geographical location** in the CERN campus
- **+300M** of files processed every day
- **+4PB** of files backed up
- **~40k** backup jobs



cback backup system

Architecture

- Composed by **stateless** agents (backup, restore, prune, verify and sync)
- They share the state (i.e. the jobs to run) in a **central database**



cback backup system

Architecture

- Agents pick a random job from a **pending queue** and run it (**asynchronous**)
- The operator can run a job **synchronously**
- After successfully run a job, the agent enqueues it in a **completed queue**
- A **switch** agent move a job from the completed to the pending queue periodically
 - For example every 24h after the job finished to run

Jobs in cback (I)

Backup

- Backup is executed per user
- Only when needed (only for *eosxd* and *cephfs*)
- In general a backup starts ~24h after it finished
 - The start time of a backup job could vary with the time
- A **desired start time** can be specified (time of the day in which run a job)
- **Node pinning**

Restore

- **Restore** data from the repository
- **Full / partial** restore
- Restore can be done **locally** or in the **backup source**
- Used by internal **durability** processes

Jobs in cback (II)

Verify

- **Restore** locally a repository and run a **consistency check** using built-in restic check feature
- When the backup source is EOS, a second phase verification consists of an **integrity check** using a namespace dump

Prune

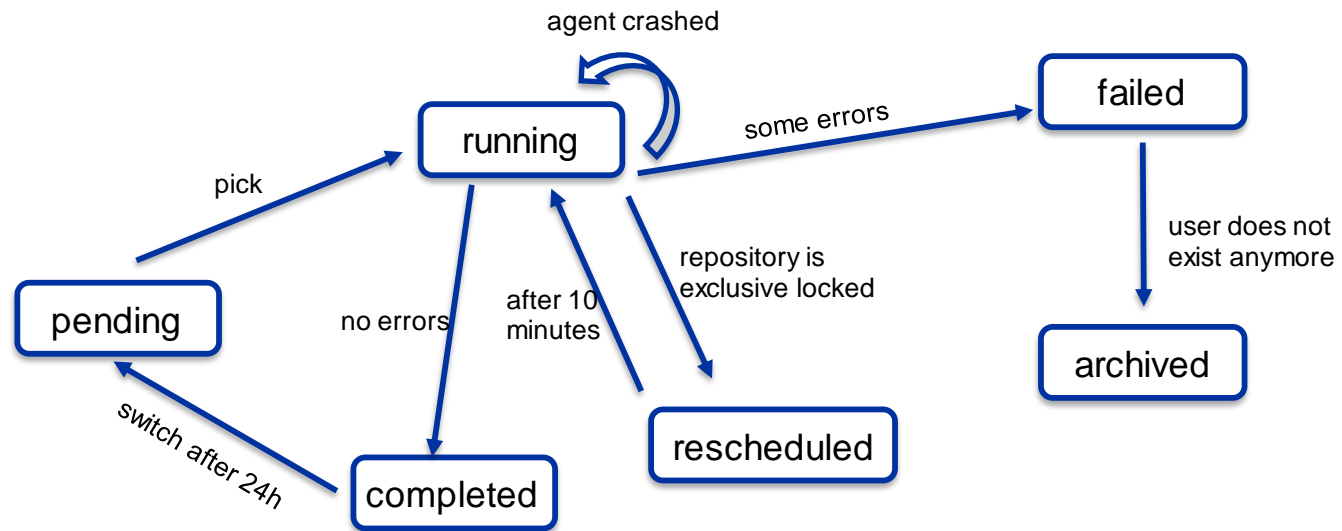
- Ensures snapshots are kept according to a **retention policy**
 - 7 daily snapshots, 5 weekly snapshots and 6 monthly snapshots
- Archived backups (user left CERN, a project is removed)
 - only last snapshot is kept

Jobs in cback (III)

Sync

- Allows a **replication** of a repository, kept aligned with the original repository when updated
 - Useful for example to have replicated backups into different geographical locations
- Allows a **migration** of a repository
 - For example between different s3 regions

Job life cycle



Using cback for different services

- cback is used to backup **several services**:
 - CERNBox (EOS)
 - Manila shares / CVMFS (CephFS)
 - EOS Media websites (EOS)
 - (+ more in the future)
- Having different instances of cback is *expensive*:
 - **Resources**: different machines per service means replication of resources, that in some cases could not be fully exploited.
 - **Managing**: an update of cback requires an update in all the instances. Different configurations are spread into different machines.

Groups

Overview

- cback introduces the concept of **groups**
- Every group could be seen as a virtual cback running inside cback
 - With a **configuration** completely **separated** by the other groups
- **Jobs** are associated to a group
- Every agent will still run a job no matter which group belongs to
- **Enable/disable** an agent in a **specific group**
- Virtually groups could be **infinite**
- For example, CERNBox, EOS Media websites and the Manila Shares could be backed up by the same cback instance

What's next?

- **System hardening, two-factor authentication** to access machines running the agents
- Expose cback functionalities through a **portal**
 - Integrated with an authentication manager (like Keycloak)
 - Implement RBAC
 - Allows the interoperability with other softwares (like CERNBox)
 - Expose **self-service restore** to CERNBox users
 - Expose the **navigation of snapshots** to CERNBox users
 - Durability procedures could interact with cback using the APIs exposed by the portal, instead of ssh-ing to a machine and run a command

What's next?

- **Optimization**
 - The majority of files are unmodified
 - Currently restic scans the whole folder
 - Use the xattrs and only read the updated folders/files ([restic/restic#3648](#))
 - This also requires to merge the new snapshot with the previous
- **Cold storage**
- Release the tool to the community



CERNBox backup evolution

A special thanks to:

Roberto Valverde Cameselle
Hugo Gonzalez Labrador
Giuseppe Lo Presti

Gianmaria Del Monte – IT-ST
gianmaria.del.monte@cern.ch