Globally Distributed Software Defined Storage (proposal)

Sergey Khoruzhnikov, Vladimir Gudinin, Oleg Sadov, Andrey Shevel, Arsen Kaikyanov, Oleg Lazo, Anatoly Greshnik

23rd International Conference on Computing in High Energy and Nuclear Physics, hosted by SLAC and LBNL, Fall 2016
Presenter: Andrey Y Shevel (Andrey.Shevel@pnpi.spb.ru, Shevel_A_Y@itmo.ru)

ITMO University, S.Petersburg (Russia)
National Research Centre "Kurchatov Institute" PETERSBURG NUCLEAR PHYSICS INSTITUTE

Technical details of GDSSD

- Important features:
  - Data storage and Data transfer
  - Reliability: data replication, erasure coding
  - Reduce the volume: data compression
  - Security: Data encryption, ACL
  - GDSSD Web portal and GDSSD CLI
  - Network architecture.
  - Caching, Tiering
  - Automatic storage deployment by user request.

Network aspects on GDSSD

- First of all we have to keep in mind the CAP theorem:
  - Theoretically it is NOT possible to guarantee all below requirements at the same time.
  - Consistency
  - Availability
  - Partitioning

Similar (in some aspects) developments

- Project CalDB at University of Michigan - https://indico.cern.ch/event/496836/contributions/1143627/
- http://less.cern.ch
- Opencloud.org, NewCloud.com

Basic assumptions on GDSSD

- GSSD consists of several groups of storage servers located in geographically different regions.
- Groups of servers are connected by a number of parallel virtual data links.
- Data links have different features: speed, price, encryption type (disk, SAN, LAN), etc.
- Data links are configured with SDS.
- Client can access a number of operations:
  - Create, Upgrade, Downgrade, Delete, Rebalance, Migration, etc. instance of virtual Storage Volumes over GDSSD.
- The instance might be created with:
  - Data links configuration requirements
  - Interfacing data format
  - Interface data format instance of Storage Volumes.

Examples for SLA

- Specific type of Data Encryption:
- Specific type of Data Compression.
- On one specific Data Center (DC) or on many DCs with specific types of Data Links.
- Type of backend: CEPH, SWIFT, EOS, etc.

Development process consideration

- During implementation GDSSD the project working repository is strongly required. The project working repository must be done only with the working repository.
- Undergraduates the working repository is separate specific activity.
- Several existing SDD systems might be considered as back-end: SWIFT, CEPH, EOS.
- GDSSD is under testing now as back-end for the proposal.
- Tech implementation with docker (under intensive testing now).
- Yes, we are looking for support.

Main features of SDS

- Software Defined Storage should include:
  - Automation – Simplified management that reduces the cost of maintaining the storage infrastructure.
  - Standard interfaces – APIs for the management, provisioning and maintenance of storage resources and services.
  - Virtualized Data Path – Block, File and Object interfaces that support applications written to these interfaces.
  - Scalability – The ability to scale the infrastructure without disruption to availability or performance.

References

- [Mastrovito - Team of Australia (literature technologies) / ACSE 2014, Project (for) Submission: Ac database as a service/ida/1088/754-3936/46384/46384.pdf]
- Why so Small? Ceph based storage at the RAL Tier-1
  - https://indico.cern.ch/event/496836/contributions/1143627/
- [Analyses of Six Distributed File Systems – File data]
  - https://indico.cern.ch/event/496836/contributions/1143627/pdfs
- [A brief introduction to Distributed File Systems]
- [ShareFS is a fault-tolerant distributed file system for all storage needs]
  - http://www.cern.ch/itf
- [Software Defined Storage (OpenSD) is distributed, scalable, fault-tolerant and highly available file system]
  - https://sharefs.cern.ch/