21st International Conference on Computing in High Energy and Nuclear Physics **CHEP2015** Okinawa Japan: April 13 - 17, 2015

Contribution ID: **535** Type: **poster presentation**

Data Integrity for Silent Data Corruption in Gfarm File System

Files in storage are often corrupted silently without any explicit error. This is typically due to file system software bug, RAID controller firmware bug, and some other reasons. Most critical issue is damaged data is read without any error. Although there are several mechanisms to detect data corruption in different layers such as ECC in disk and memory and TCP checksum, the data may be damaged. To cope with the silent data corruption, the file system level detection is effective. Btrfs and ZFS have a mechanism to detect it by adding checksum in each block. However, data replication is often required to correct the damaged data, which may waste storage capacity in local file system since it is required only for data integrity.

Gfarm file system is a distributed file system that federates storages among several institutions in wide area. Large installations include Japan Lattice Data Grid (JLDG) with 4PB storage capacity in 9 storage sites, and HPCI shared storage with 20PB storage capacity in 3 storage sites. It has file replicas to improve access performance from distant clients, and also to improve fault tolerance. The number and the locations of file replicas are specified by an extended attribute of a directory or a file.

We design the data integrity feature in Gfarm file system by automatically calculating digest like md5 or sha256 when accessing files. The file digest is calculatd at a storage node before writing to a storage when a file is created, and managed in file system metadata. It can detect data corruption even when writing to storage. The digest is also calculated when reading a file. After reading entire data of file, the read system call returns input/output error when the digest mismatches. This ensures corrupted data cannot be read. When creating a file replica, this digest check is also performed. To cope with data corruption during the network transfer from a client, it also supports digest calculation at client side to ensure end-to-end data integrity.

This paper describes a design and implementation of data integrity feature in Gfarm file system. Due to native and required feature of file replicas in Gfarm file system, the data integrity can be supported without any waste storage capacity. To reduce additional overhead for calculating digest, the digest is only calculated when accessing files sequentially. This access pattern is the most typical case in Gfarm file system. For files that is created by random access write, the digest is calculated when creating the file replica. This design enables a minimum additional overhead and enough data integrity support.

Author: TATEBE, Osamu (University of Tsukuba)

Co-author: YOSHIE, Tomoteru (U)

Presenter: TATEBE, Osamu (University of Tsukuba)

Track Classification: Track3: Data store and access