Erasure encoding overview

Abstract

In the following document, several common erasure encodings are described together with their advantages and disadvantages and open-source implementations

Basic concepts

The main idea of nearly every erasure coding is to

1. Separate initial data into blocks of size k called words (or messages) W
2. Compute a certain function f(W) which has output of size m called parity
3. Store k + m = n items of data, providing the resilience to failure of at most m nodes, containing data, provided each piece of data is put on a separate node

Advantages

1. Durability: failure up to m discs with data will not lead to data loss
2. Given x blocks of information it takes (x/k)*m storage space instead of (t - 1)*x additional space as opposed to t-replication

Disadvantages

1. Needs computational power in order to compute parity
2. In order to get initial block of data the decoding process needs to be performed
3. After writing to file parity needs to be at least partially recomputed

Available codings

Reed-Solomon (RS)

- Calculates f using matrix multiplication over GF (2^w)
- Decoding involves matrix inversion and multiplication over GF (2^w)

Cauchy-Reed-Solomon (CRS)

- As opposed to RS uses only addition over GF (2^w) (XOR) which makes it much faster than RS according to [4]

Linux RAID-6

- Uses only multiplications by 1 (id) and 2 (<<) and XORs, according to [4] is more reliable than CRS (its encoding/decoding speed is more consistent)

EVENODD

- Doesn't use matrices operations
- High costs of recalculating parity after updates

RDP

- Achieves the theoretical minimum of XORs required for encoding and decoding
- Has the same update penalty as EVENODD

Open-source implementations

1. fecpp - RS
2. Jerasure - RS, CRS, RAID-6

Current state of events

Currently erasure coding using Jerasure is included in ceph and the alpha testing will commence as soon as tests are written, according to [3]

Links

1. Tutorial on Erasure Coding for Storage Applications - Huge tutorial on erasure coding, includes overview of nearly all used codings and its implementations
2. ERASURE CODE DEVELOPER NOTES @ Ceph - documentation on coding, implemented in Ceph so far
3. Ceph tracker issue 4929 - Ceph tracker ticket on implementation of erasure coding
4. A Performance Evaluation and Examination of Open-Source Erasure Coding Libraries For Storage - article on encoding speeds of various algorithms circa 2009