CEPHFS-JOURNAL-TOOL

PURPOSE

If a CephFS journal has become damaged, expert intervention may be required to restore the filesystem to a working state.

The `cephfs-journal-tool` utility provides functionality to aid experts in examining, modifying, and extracting data from journals.

Warning: This tool is **dangerous** because it directly modifies internal data structures of the filesystem. Make backups, be careful, and seek expert advice. If you are unsure, do not run this tool.

SYNTAX

```
cephfs-journal-tool journal <inspect|import|export|reset>
cephfs-journal-tool header <get|set>
cephfs-journal-tool event <get|splice|apply> [filter] <list|json|summary>
```

The tool operates in three modes: `journal`, `header` and `event`, meaning the whole journal, the header, and the events within the journal respectively.

JOURNAL MODE

This should be your starting point to assess the state of a journal.

- `inspect` reports on the health of the journal. This will identify any missing objects or corruption in the stored journal. Note that this does not identify inconsistencies in the events themselves, just that events are present and can be decoded.
- `import` and `export` read and write binary dumps of the journal in a sparse file format. Pass the filename as the last argument. The export operation may not work reliably for journals which are damaged (missing objects).
- `reset` truncates a journal, discarding any information within it.

EXAMPLE: JOURNAL INSPECT

```
# cephfs-journal-tool journal inspect
Overall journal integrity: DAMAGED
Objects missing:
  0x1
Corrupt regions:
  0x400000-ffffffffffffffff
```

EXAMPLE: JOURNAL IMPORT/EXPORT

```
# cephfs-journal-tool journal export myjournal.bin
journal is 4194304~80643
read 80643 bytes at offset 4194304
wrote 80643 bytes at offset 4194304 to myjournal.bin
NOTE: this is a _sparse_ file; you can
    $ tar cSzf myjournal.bin.tgz myjournal.bin
      to efficiently compress it while preserving sparseness.

# cephfs-journal-tool journal import myjournal.bin
undump myjournal.bin
start 4194304 len 80643
writing header 200.00000000
 writing 4194304~80643
done.
```

Note: It is wise to use the `journal export <backup file>` command to make a journal backup before any further manipulation.

HEADER MODE

- `get` outputs the current content of the journal header
- `set` modifies an attribute of the header. Allowed attributes are `trimmed_pos`, `expire_pos` and `write_pos`.

EXAMPLE: HEADER GET/SET

```
# cephfs-journal-tool header get
{ "magic": "ceph fs volume v011",
  "write_pos": 4274947,
  "expire_pos": 4194304,
  "trimmed_pos": 4194303,
  "layout": { "stripe_unit": 4194304,
      "stripe_count": 4194304,
      "object_size": 4194304,
      "cas_hash": 4194304,
      "object_stripe_unit": 4194304,
      "pg_pool": 4194304}}

# cephfs-journal-tool header set trimmed_pos 4194303
Updating trimmed_pos 0x400000 -> 0x3fffff
Successfully updated header.
```

EVENT MODE

Event mode allows detailed examination and manipulation of the contents of the journal. Event mode can operate on all events in the journal, or filters may be applied.

The arguments following `cephfs-journal-tool event` consist of an action, optional filter parameters, and an output mode:

```
cephfs-journal-tool event <action> [filter] <output>
```

Actions:

- `get` read the events from the log
- `splice` erase events or regions in the journal
- `apply` extract filesystem metadata from events and attempt to apply it to the metadata store.

Filtering:

- `--range <int begin>..[int end]` only include events within the range begin (inclusive) to end (exclusive)
- `--path <path substring>` only include events referring to metadata containing the specified string
- `--inode <int>` only include events referring to metadata containing the specified string
- `--type <type string>` only include events of this type
- `--frag <ino>[.frag id]` only include events referring to this directory fragment
- `--dname <string>` only include events referring to this named dentry within a directory fragment (may only be used in conjunction with `--frag`
- `--client <int>` only include events from this client session ID

Filters may be combined on an AND basis (i.e. only the intersection of events from each filter).

Output modes:

- `binary`: write each event as a binary file, within a folder whose name is controlled by `--path`
- `json`: write all events to a single file, as a JSON serialized list of objects
- `summary`: write a human readable summary of the events read to standard out
- `list`: write a human readable terse listing of the type of each event, and which file paths the event affects.

EXAMPLE: EVENT MODE

```
# cephfs-journal-tool event get json --path output.json
Wrote output to JSON file 'output.json'

# cephfs-journal-tool event get summary
Events by type:
  NOOP: 2
  OPEN: 2
  SESSION: 2
  SUBTREEMAP: 1
  UPDATE: 43

# cephfs-journal-tool event get list
0x400000 SUBTREEMAP:  ()
0x400308 SESSION:  ()
0x4003de UPDATE:  (setattr)
  /
0x40068b UPDATE:  (mkdir)
  diralpha
0x400d1b UPDATE:  (mkdir)
  diralpha/filealpha1
0x401666 UPDATE:  (unlink_local)
  stray0/10000000001
  diralpha/filealpha1
0x40228d UPDATE:  (unlink_local)
  diralpha
  stray0/10000000000
0x402bf9 UPDATE:  (scatter_writebehind)
  stray0
0x403150 UPDATE:  (mkdir)
  dirbravo
0x4037e0 UPDATE:  (openc)
  dirbravo/.filebravo1.swp
0x404032 UPDATE:  (openc)
  dirbravo/.filebravo1.swpx

# cephfs-journal-tool event get --path /filebravo1 list
0x40785a UPDATE:  (openc)
  dirbravo/filebravo1
0x4103ee UPDATE:  (cap update)
  dirbravo/filebravo1

# cephfs-journal-tool event splice --range 0x40f754..0x410bf1 summary
Events by type:
  OPEN: 1
  UPDATE: 2

# cephfs-journal-tool event apply --range 0x410bf1.. summary
Events by type:
  NOOP: 1
  SESSION: 1
  UPDATE: 9

# cephfs-journal-tool event get --inode=1099511627776 list
0x40068b UPDATE:  (mkdir)
  diralpha
0x400d1b UPDATE:  (mkdir)
  diralpha/filealpha1
0x401666 UPDATE:  (unlink_local)
  stray0/10000000001
  diralpha/filealpha1
0x40228d UPDATE:  (unlink_local)
  diralpha
  stray0/10000000000

# cephfs-journal-tool event get --frag=1099511627776 --dname=filealpha1 list
0x400d1b UPDATE:  (mkdir)
  diralpha/filealpha1
0x401666 UPDATE:  (unlink_local)
  stray0/10000000001
  diralpha/filealpha1

# cephfs-journal-tool event get binary --path bin_events
```

```
Wrote output to binary files in directory 'bin_events'
```