The Linux pNFS Server VFS API

WIP Discussion

Benny Halevy <bhalevy@panasas.com>
Connectathon 2009

Talk Outline

- Requirements
- VFS Extensions
- Issues
- Open Discussion

Requirements

- Exporting File Systems over pNFS:
  - Get, Commit, and Return Layouts
  - Get Device List, Device Info
  - Callbacks: Recall Layouts, Recall Any, Device Notifications.
    - May be initiated from the Filesystem rather than the nfs server.

More Requirements

- Support multiple layout types
  - Per- file system? (module) or per-superblock (fsid).
  - Provide Layout-type policies.
- Files layout DS <-> MDS global state hooks.

Legacy export_operations

- int (***encode_fh**)(dentry, fh, max_len, connectable);
- dentry * (***fh_to_dentry**)(super_block, fid, fh_len, fh_type);
- dentry * (***fh_to_parent**)(super_block, fid, fh_len, fh_type);
- int (***get_name**)(dentry *parent, char *name, dentry *child);
- dentry * (***get_parent**)(dentry *child);

pNFS VFS Extensions

- int (***layout_type**)(super_block *);
  - Is pNFS Supported for the exported super_block?
  - If so, what's its layout-type.

Layout Ops

- int (***layout_get**) (inode *, pnfs_layoutget *);
- int (***layout_commit**) (inode *, pnfs_layoutcommit *);
- int (***layout_return**) (inode *, pnfs_layoutreturn *);

Device Ops

- int (*__get_device_info__) (super_block*, pnfs_devinfo *);
- int (*__get_device_iter__) (super_block *, pnfs_deviter *);

VFS Callbacks

- int (***cb_layout_recall**) (super_block *, inode *,
  pnfs_cb_layout *);
  int (***cb_device_notify**) (super_block *,
  pnfs_cb_devnotify *);
- Vectors provided by nfsd.
- Callbacks issued by file system.

Policies

- int (*__can_merge_layouts__)(u32 layout_type);

Files layout-type

- void (***get_verifier**) (struct super_block *sb, u32 *p);
  int (***cb_get_state**) (struct super_block *sb, void *state);
  int (***get_state**) (struct inode *inode, void *fh, void *state);
  int (***cb_change_state**) (void *p);

Issues

- We're spamming struct export_operations.
- Stackable filesystems override methods
- Long-living stable enterprise distributions
- Inverted module dependency on callback path.
  - Callbacks may be issued for graceful shutdown / umount.

Even More Requirements?

- Implement MDS in user mode.
  - Spnfs today
  - Extend linux FUSE in the future?
- pNFS-enabled file system interface for user mode apps (E.g. Hadoop HDFS)

Open Discussion

Backup Slides

Layout get

- int (*layout_get) (inode, pnfs_layoutget_arg *);

```
struct pnfs_layoutget_arg {
   minlength;   /* request */
   func;        /* request */
   fsid;        /* request */
   fh;                      /* request/response */
   layout_seg;              /* request/response */
   pnfs_xdr_info  xdr;      /* request/response */
   return_on_close;         /* response */
};
```

- Opaque layout-type specific buffers passed down to the filesystem.
- The generic layer still manages a generic layout (segments) cache.

Layout Commit

- int (*layout_commit) (inode,
  pnfs_layoutcommit *);

```
struct nfsd4_pnfs_layoutcommit {
   layout_seg;     /* request */
   lc_reclaim;     /* request */
   lc_newoffset;   /* request */
   lc_last_wr_offs;/* request */
   lc_mtime;       /* request */
   lc_stateid;     /* request */
   lc_up_len;      /* layout-update length */
   void *lc_up_layout;
   lc_size_chg; /* boolean for response */
   lc_newsize;  /* response */
};
```

Layout Return

- int (*layout_return) (inode,
                                    pnfs_layoutreturn *);

```
struct nfsd4_pnfs_layoutreturn {
   lr_return_type;    /* request */
   lr_seg;            /* request */
   lr_reclaim;        /* request */
   lr_flags;          /* internal */
   lr_stateid;        /* request/response */
   lrf_body_len;      /* lo-type specific */
   void *lrf_body;
   lrstate_present;        /* response */
};
```

Get Device Info/List

- int (*get_device_info) (super_block *, struct pnfs_devinfo_arg *arg);

```
struct pnfs_devinfo_arg {
   type;          /* request */
   devid;         /* request */
   notify_types;      /* request/response */
   xdr;               /* request/response */
   pnfs_encodedev_t func;   /* request */
};
```

- int (*get_device_iter) (super_block *, struct pnfs_deviter_arg *arg);
-