CONFIG-KEY LAYOUT

config-key is a general-purpose key/value storage service offered by the mons. Generally speaking, you can put whatever you want there. Current in-tree users should be captured here with their key layout schema.

OSD DM-CRYPT KEYS

Key:

```
dm-crypt/osd/$OSD_UUID/luks = <json string>
```

The JSON payload has the form:

```
{ "dm-crypt": <secret> }
```

where the secret is a base64 encoded LUKS key.

Created by the 'osd new' command (see OSDMonitor.cc).

Consumed by ceph-disk, ceph-volume, and similar tools. Normally access to the dm-crypt/osd/$OSD_UUID prefix is allowed by a client.osd-lockbox.$OSD_UUID cephx key, such that only the appropriate host can retrieve the LUKS key (which in turn decrypts the actual raw key, also stored on the device itself).

CEPH-MGR MODULES

The convention for keys is:

```
mgr/$MODULE/$option = $value
```

or:

```
mgr/$MODULE/$MGRID/$option = $value
```

For example,:

```
mgr/dashboard/server_port = 80
mgr/dashboard/foo/server_addr = 1.2.3.4
mgr/dashboard/bar/server_addr = 1.2.3.5
```

CONFIGURATION

Configuration options for clients and daemons are also stored in config-key.

Keys take the form:

```
config/$option = $value
config/$type/$option = $value
config/$type.$id/$option = $value
config/$type.$id/$mask[/$mask2...]/$option = $value
```

Where

- *type* is a daemon type (*osd*, *mon*, *mds*, *mgr*, *client*)
- *id* is a daemon id (e.g., *0*, *foo*), such that *$type.$id* is something like *osd.123* or *mds.foo*)
- *mask* restricts who the option applies to, and can take two forms:

1. *$crush_type:$crush_value*. For example, *rack:foorack*
2. *class:$classname*, in reference to CRUSH device classes (e.g., *ssd*)