store a copy of the /usr/share/dict/words file at the root of the file system. Unmount and unmap the device when done.

8.1. Map the `data` image in the `rbd` pool using the kernel RBD client.

```
[admin@clienta ~]$ sudo rbd map --pool rbd data
/dev/rbd0
```

8.2. Format the /dev/rbd0 device with an XFS file system and mount the file system on the `/mnt/data` directory.

```
[admin@clienta ~]$ sudo mkfs.xfs /dev/rbd0
meta-data=/dev/rbd0              isize=512    agcount=8, agsize=4096 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=1, sparse=1, rmapbt=0
         =                       reflink=1
data     =                       bsize=4096   blocks=32768, imaxpct=25
         =                       sunit=16     swidth=16 blks
naming   =version 2              bsize=4096   ascii-ci=0, ftype=1
log      =internal log           bsize=4096   blocks=1872, version=2
         =                       sectsz=512   sunit=16 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
Discarding blocks...Done.
[admin@clienta ~]$ sudo mount /dev/rbd0 /mnt/data
```

8.3. Copy the `/usr/share/dict/words` file to the root of the file system, `/mnt/data`. List the content to verify the copy.

```
[admin@clienta ~]$ sudo cp /usr/share/dict/words /mnt/data/
[admin@clienta ~]$ ls /mnt/data/
words
```

8.4. Unmount and unmap the `/dev/rbd0` device.

```
[admin@clienta ~]$ sudo umount /dev/rbd0
[admin@clienta ~]$ sudo rbd unmap --pool rbd data
```

9. In the production cluster, create a snapshot called `beforeprod` of the RBD image `data`. Create a clone called `prod1` from the snapshot called `beforeprod`.

9.1. In the production cluster, use `sudo` to run the `cephadm` shell. Create a snapshot called `beforeprod` of the RBD image `data` in the `rbd` pool.

```
[admin@clienta ~]$ sudo cephadm shell
...output omitted...
[ceph: root@clienta /]# rbd snap create rbd/data@beforeprod
Creating snap: 100% complete...done.
```

9.2. Verify the snapshot by listing the snapshots of the `data` RBD image in the `rbd` pool.