Disaster recovery

replace NVM device

Only 1 NVM slot available, so idea is to copy nvm to hdd and then restore it on new nvm device.

Stop CEPH:

```
systemctl stop ceph.target
systemctl stop ceph-osd.target
systemctl stop ceph-mgr.target
systemctl stop ceph-mon.target
systemctl stop ceph-mds.target
systemctl stop ceph-crash.service
```

Backup partition layout

```
sgdisk -b nvm.sgdisk /dev/nvme0n1
sgdisk -p /dev/nvme0n1
```

Move ZFS nvmpool to hdds:

```
zfs destroy hddpool/nvmtemp
zfs create -s -b 8192 -V 387.8G hddpool/nvmtemp  # not block size was forced
to match existing device

ls -l /dev/zvol/hddpool/nvmtemp
lrwxrwxrwx 1 root root 11 01-15 11:00 /dev/zvol/hddpool/nvmtemp ->
../../zd192

zpool attach nvmpool 7b375b69-3ef9-c94b-bab5-ef68f13df47c /dev/zd192
```

And `nvmpool` resilvering will begin. Observe it with `zpool status nvmpool 1`

Remove NVM from `nvmpool`:

```
zpool detach nvmpool 7b375b69-3ef9-c94b-bab5-ef68f13df47c
```

Remove all ZILS, L2ARCs and swap:

```
swapoff -a
vi /etc/fstab

zpool remove hddpool <ZIL DEVICE>
zpool remove hddpool <L2ARC DEVICE>
zpool remove rpool <L2ARC DEVICE>
```

CEPH OSD will be created from scratch to force to rebuild OSD DB (which can be too big due to

metadata bug from previous version of CEPH)

Replace NVM.

Recreate partitions or restore from backup

```
sgdisk -l nvm.sgdisk /dev/nvme0n1
```

- swap
- rpool_zil
- hddpool_zil
- hddpool_l2arc
- ceph_db (for 4GB ceph OSD create 4096MB+4MB)

Add ZILs and L2ARCs.

Start `nvmpool`:

```
zpool import nvmpool
```

Move `nvmpool` to new NVM partition:

```
zpool attach nvmpool zd16 426718f1-1b1e-40c0-a6e2-1332fe5c3f2c
zpool detach nvmpool zd16
```

Replace rpool device

Proxmox rpool ZFS is located on 3rd partition (1st is Grub BOOT, 2nd is EFI, 3rd is ZFS). To replace failed device it is needed to replicate partition layout:

With new device of greater or equal size, simple replicate partitions:

```
# replicate layout from SDA to SDB
sgdisk /dev/sda -R /dev/sdb
# generate new UUIDs:
sgdisk -G /dev/sdb
```

To replicate layout on smaller device, need manually create partitions:

```
sgdisk -p /dev/sda

Number  Start (sector)    End (sector)  Size       Code  Name
   1              34            2047   1007.0 KiB  EF02
   2            2048         1050623   512.0 MiB   EF00
   3         1050624       976773134   465.3 GiB   BF01

sgdisk --clear /dev/sdb
sgdisk /dev/sdb -a1 --new 1:34:2047      -t0:EF02
sgdisk /dev/sdb     --new 2:2048:1050623 -t0:EF00
```

```
sgdisk /dev/sdb     --new 3:1050624      -t0:BF01
```

Restore bootloader:

```
proxmox-boot-tool format /dev/sdb2
proxmox-boot-tool init /dev/sdb2
proxmox-boot-tool clean
```

```
zpool attach rpool ata-SPCC_Solid_State_Disk_XXXXXXXXXXXX-part3
/dev/disk/by-id/ata-SSDPR-CL100-120-G3_XXXXXXXX-part3
zpool offline rpool ata-SSDPR-CX400-128-G2_XXXXXXXXX-part3
zpool detach rpool ata-SSDPR-CX400-128-G2_XXXXXXXXX-part3
```

Migrate VM from dead node

Simply move config file:

```
mv ./nodes/pve3/qemu-server/366.conf ./nodes/pve5/qemu-server/
```

If move is not possible (no quorum in cluster), simply reduce expected votes to 1:

```
pvecm e 1
```

Transfer needed storage. From source storage node pve3 send volume to pve5:

```
zfs send rpool2/data/vm-366-disk-0 | ssh pve5 zfs recv -d rpool
zfs send rpool2/data/vm-366-disk-2 | ssh pve5 zfs recv -d rpool
```

reinstall node

Remember to clean any additional device partition belonging to `rpool` (i.e. ZIL). During Proxmox first startup ZFS detects that there are two `rpool` in system and stops requiring importing by its numerical id.

Install fresh Proxmox. Create common cluster-wide mountpoints to local storage. Copy all zfs datasets from backup ZFS pool:

```
zfs send rpool2/data/vm-708-disk-0 | zfs recv -d rpool
...
```

For CT volumes it getting more complicated:

```
root@pve3:~# zfs send rpool2/data/subvol-806-disk-0 | zfs recv -d rpool
warning: cannot send 'rpool2/data/subvol-806-disk-0': target is busy; if a
filesystem, it must not be mounted
cannot receive: failed to read from stream
```

Reason of problem is that SOURCE is mounted. Solution:

```
zfs set canmount=off rpool2/data/subvol-806-disk-0
```

Try to join to cluster. From new (reinstalled) node pve3 join to IP of any existing node. Needs to use –force switch, because pve3 node was previously in cluster.

```
root@pve3:~# pvecm add 192.168.28.235 --force

Please enter superuser (root) password for '192.168.28.235': ************
Establishing API connection with host '192.168.28.235'
The authenticity of host '192.168.28.235' can't be established.
X509 SHA256 key fingerprint is
D2:68:21:D7:43:6D:BA:4D:EB:C6:32:DD:2C:72:6E:5B:6D:1A:2D:DB:82:EC:E6:41:72:4
6:6B:E6:B1:BF:94:84.
Are you sure you want to continue connecting (yes/no)? yes
Login succeeded.
check cluster join API version
No cluster network links passed explicitly, fallback to local node IP
'192.168.28.233'
Request addition of this node
Join request OK, finishing setup locally
stopping pve-cluster service
backup old database to '/var/lib/pve-
cluster/backup/config-1621353318.sql.gz'
waiting for quorum...OK
(re)generate node files
generate new node certificate
merge authorized SSH keys and known hosts
generated new node certificate, restart pveproxy and pvedaemon services
successfully added node 'pve3' to cluster.
```