

Proxmox Installation mit Cluster und CEPH

Admin Guide als PDF

- [coming soon](#)

Proxmox auf den Nodes installieren

Bei der Festplattenzuweisung darauf achten das man bei der Festplatte an sich nicht alles angibt sondern nur soviel wie man für swap und root (samt iso files bei N1) benötigt. Für die Nodes ohne die ISO Platte sind das 18GB, die mit der Iso Platte bekommt 48GB

System (root) N1 und N5 40GB

System (root) 10GB

minimal N1 und N5 2Gb

minimal 1GB

Ram 8GB

Data 0GB

n1-pve-cluster.tuxnet.local 192.168.179.5

n2-pve-cluster.tuxnet.local 192.168.179.6

n3-pve-cluster.tuxnet.local 192.168.179.7

n4-pve-cluster.tuxnet.local 192.168.179.8

n5-pve-cluster.tuxnet.local 192.168.179.9

Beispiel für Netzwerk

Konfiguration für Proxmox Cluster Netzwerk, Ceph Clusternetzwerk und normales Bridgenetzwerk.
/etc/network/interfaces

```
auto lo
iface lo inet loopback

iface eno1 inet manual

auto vmbr0
iface vmbr0 inet static
    address 192.168.179.6
    netmask 255.255.255.0
    gateway 192.168.179.1
    bridge_ports eno1
    bridge_stp off
    bridge_fd 0
```

```
# corosync cluster network
auto eno1.321
iface eno1.321 inet static
    address 10.32.1.5
    netmask 255.255.255.0

# ceph storage cluster network
auto eno1.322
iface eno1.322 inet static
    address 10.32.2.5
    netmask 255.255.255.0
```

Proxmox auf den Nodes einrichten

```
apt-get update
apt-get dist-upgrade
apt-get install vim
```

/etc/hosts auf jeder Node anpassen so das alle Nodes drin stehen.

```
127.0.0.1 localhost.localdomain localhost
192.168.179.5 n1-pve-cluster.tuxnet.local n1-pve-cluster pvelocalhost
192.168.179.6 n2-pve-cluster.tuxnet.local n2-pve-cluster
192.168.179.7 n3-pve-cluster.tuxnet.local n3-pve-cluster
192.168.179.8 n4-pve-cluster.tuxnet.local n4-pve-cluster
192.168.179.9 n5-pve-cluster.tuxnet.local n5-pve-cluster

# corosync cluster network
10.32.1.5 n1-corosync-cluster.tuxnet.local n1-corosync-cluster
10.32.1.6 n2-corosync-cluster.tuxnet.local n2-corosync-cluster
10.32.1.7 n3-corosync-cluster.tuxnet.local n3-corosync-cluster
10.32.1.8 n4-corosync-cluster.tuxnet.local n4-corosync-cluster
10.32.1.9 n5-corosync-cluster.tuxnet.local n5-corosync-cluster

# ceph storage cluster network
10.32.2.5 n1-ceph-cluster.tuxnet.local n1-ceph-cluster
10.32.2.6 n2-ceph-cluster.tuxnet.local n2-ceph-cluster
10.32.2.7 n3-ceph-cluster.tuxnet.local n3-ceph-cluster
10.32.2.8 n4-ceph-cluster.tuxnet.local n4-ceph-cluster
10.32.2.9 n5-ceph-cluster.tuxnet.local n5-ceph-cluster
```

```
# The following lines are desirable for IPv6 capable hosts
```

```
::1      ip6-localhost ip6-loopback
fe00::0  ip6-localnet
ff00::0  ip6-mcastprefix
ff02::1  ip6-allnodes
ff02::2  ip6-allrouters
ff02::3  ip6-allhosts
```

Cluster auf erster Clusternode initialisieren

pvecm create <clustername> -bindnet0_addr ip-node -ring0_addr hostname-coro-cluster # wie in der /etc/hosts steht

pvecm create PVE01-CLUSTER -bindnet0_addr 10.32.1.5 -ring0_addr n1-corosync-cluster

Dienste neu starten

```
root@n1-pve-cluster:~# systemctl restart corosync.service
root@n1-pve-cluster:~# systemctl restart pve-cluster.service
root@n1-pve-cluster:~# systemctl restart pvedaemon.service
root@n1-pve-cluster:~# systemctl restart pveproxy.service
```

Danach schauen ob alles ok ist pvecm status

Dann auf allen anderen Nodes die Nodes dem Cluster hinzufügen

```
pvecm add 10.32.1.5 -ring0_addr n2-corosync-cluster # n2-corosync-cluster
wie in der lokalen /etc/hosts für den lokalen Namen steht
10.32.1.5 = ip eines Clustermembers
```

CEPH Cluster einrichten

Den noch freien Speicherplatz der SSD auf allen Nodes einer Partition zuordnen. Partitionstyp ist ceph-osd.

Packete auf jeder Node installieren von der luminous Version

```
pveceph install --version luminous
```

```
pveceph init --network 10.32.2.0/24
```

auf der ersten Node ausführen.

Danach wird auf jeder Node der Monitor erstellt

```
pveceph createmon
```

Proxmox erlaubt normalerweise nicht die Angabe einer Partitionsnummer für OSD. Wie man es dennoch entsprechend einrichten kann erklärt [dieser Artikel](https://blog.cooltux.net/)

Eigenes Server Zertifikat einspielen

Erstelle ein Serverzertifikat und zeichne es gegen das rootCA gegen. Danach das Zertifikat und den Key exportieren und über das Webfrontend von Proxmox importieren. Zum Abschluss wird das Proxmox rootCA gegen unser eigenes ausgetauscht, damit vnc funktioniert. Wenn alles geklappt hat kann direkt auf dem Server das alte Zertifikat sowie das Proxmox rootCA Zertifikat gelöscht werden.

```
rm /etc/pve/local/pve-ssl.*
rm /etc/pve/pve-root-ca.pem
```

Ganz wichtig aber ist das pve-www.key nicht gelöscht werden darf, da sonst keine Anmeldung mehr über das Frontend möglich ist.

Root Partition erweitern

Unter

```
/var/lib/ceph/osd/
```

läuft der Ceph monitor und hält die Monitor Daten bereit. Diese Daten werden im Laufe der Zeit umfangreicher und nehmen entsprechend Platz ein. Die Partition sollte entsprechend groß sein. Hier am Beispiel wird die SWAP Partition verkleinert und root vergrößert. Das geht weil beide Logical Volumes in der Volume Group pve sind.

```
swapoff /dev/pve/swap
lvreduce --size 3G /dev/pve/swap
mkswap /dev/pve/swap
swapon /dev/pve/swap
lvextend -l +100%FREE /dev/pve/root
resize2fs -p /dev/pve/root
```

From:
<https://blog.cooltux.net/> - TuxNet DokuWiki

Permanent link:
<https://blog.cooltux.net/doku.php?id=it-wiki:proxmox:proxmoxinstall&rev=1563391566>

Last update: **2019/07/17 19:26**

