

Einrichtung eines Standard Proxmox4 Clusters

Voraussetzungen

In dem Beitrag wird beschrieben wie man einen Cluster (mind. 2 für Config) verbindet. Folgende Voraussetzungen müssen gegeben sein:

- Datum und Zeit synchron
- SSH über Port 22 zwischen den Nodes ist möglich
- Möchte man HA benutzen, ab 3 Nodes, sollte man überall die selbe PVE Version verwenden
- Mind. eine dizierte NIC für die Clusterkommunikation
- Eigenes Clustervlan oder eigenen PHY Switch mit eigenem Subnet
- Alle Clustermitglieder müssen auf diesem Interface das gleiche IP Netz verwenden, ob IPV4 oder IPV6 ist egal
- Alle Clustermitglieder müssen über Multicast (Corosync verwendet UDP 5404 und 5405) kommunizieren können.

Installation und Konfiguration

Als erstes installiert man auf allen Mitgliedern, in unserem Fall 2 Server, Proxmox in der gleichen Version. Wenn man eine [Subscription](#) verwendet ist es wichtig dass alle Clustermitglieder die gleiche verwenden. Hat man alles erfolgreich installiert, Updates usw. gehts an die Konfiguration.

Netzwerk einrichten

Wir gehen hier von einer Mindestkonfig von zwei NICs pro Server aus. Einmal das Homenetzwerk 192.168.0.0/24 und einmal das Proxmox Clustervlan 192.168.1.0/24. Dies kann man natürlich alles grafisch über das PVE Webinterface erledigen, oder man editiert einfach die `/etc/network/interfaces`.

Netzwerkconfig pve01:

```
auto lo
iface lo inet loopback

iface eth0 inet manual

auto vmbr0
iface vmbr0 inet static
    address 192.168.0.1
    netmask 255.255.255.0
    gateway 192.168.0.254
    bridge_ports eth0
    bridge_stp off
    bridge_fd 0
```

```
auto eth1
iface eth1 inet static
    address 192.168.1.1
    netmask 255.255.255.0
#Clusternetzwerk
```

Netzwerkconfig pve02:

```
auto lo
iface lo inet loopback

iface eth0 inet manual

auto vmbr0
iface vmbr0 inet static
    address 192.168.0.2
    netmask 255.255.255.0
    gateway 192.168.0.254
    bridge_ports eth0
    bridge_stp off
    bridge_fd 0

auto eth1
iface eth1 inet static
    address 192.168.1.2
    netmask 255.255.255.0
#Clusternetzwerk
```

Um das Neue Netzwerk zu aktivieren startet man die Server einfach durch. Man kann aber auch den Netzwerkdienst neustarten:

```
/etc/init.d/networking restart
```

Cluster generieren und erweitern

Den Cluster generiert man mit folgendem Befehl:

```
pvecmd create proxmoxcl01 -bindnet0_addr 192.168.1.0 -ring0_addr 192.168.1.1
```

Mit `pvecmd status` sieht nun das man einen Cluster mit dem Namen „proxmoxcl01“ generiert hat, dieser besteht aus einem Host. Da haben wir natürlich nicht viel davon, deshalb loggen wir uns jetzt auf dem zweiten Proxmox Server ein, dieser soll ja dem gerade generierten Cluster beitreten. Wichtig ist natürlich das sich am zweiten Host keine VM's oder sonstige Config befindet. Diese wird beim Beitritt zum Cluster überschrieben. Eingeloggt man zweiten Host, treten wir dem Cluster mit folgendem Befehl bei:

```
pvecmd add 192.168.1.1 -ring0_addr 192.168.1.2
```

Jetzt werden Zertifikate/Schlüssel usw. ausgetauscht. Auch das Proxmox [Cluster File System](#) wird [gesync](#)t. Sehen wir uns nun den Cluster an, sehen wir bereits zwei Host die synchron sind.

```
Quorum information
-----
Date:           Fri Apr 14 12:59:52 2017
Quorum provider: corosync_votequorum
Nodes:          2
Node ID:         0x00000001
Ring ID:         1/68
Quorate:         Yes
```

```
Votequorum information
-----
```

```
Expected votes:  2
Highest expected: 2
Total votes:     2
Quorum:          2
Flags:           Quorate
```

```
Membership information
-----
```

Nodeid	Votes	Name
0x00000001	1	192.168.1.1 (local)
0x00000002	1	192.168.1.2

Eine Masternode gibt es nicht, man kann also überall alles bearbeiten.

Clusterkommunikation ausfallsicher

Natürlich möchte man im Produktivbetrieb minds. zwei Interfaces für die Clusterkommunikation verwenden, falls mal eines ausfällt, oder man Wartungsarbeiten durchführen muss. Noch besser wäre wenn man die Interfaces auf zwei PHY getrennte Switches verteilt. Bei unserer Testconfig gehen wir von zwei Interfaces pro Server für die Clusterkommunikation aus. Man sollte am Switch kein Trunk/LACP oder ähnliches erstellen sondern das RRP-Protokoll von Corosync nutzen. In der Netzwerkconfig der Server muss jedes weitere Interface default hinzugefügt werden. Das ganze sieht dann so aus:

Netzwerkconfig pve01:

```
auto lo
iface lo inet loopback

iface eth0 inet manual

auto vmbr0
iface vmbr0 inet static
    address 192.168.0.1
    netmask 255.255.255.0
    gateway 192.168.0.254
```

```
bridge_ports eth0
bridge_stp off
bridge_fd 0

auto eth1
iface eth1 inet static
    address 192.168.1.1
    netmask 255.255.255.0

auto eth2
iface eth2 inet static
    address 192.168.2.1
    netmask 255.255.255.0
#Clusternetzwerk
```

Netzwerkconfig pve02:

```
auto lo
iface lo inet loopback

iface eth0 inet manual

auto vmbr0
iface vmbr0 inet static
    address 192.168.0.2
    netmask 255.255.255.0
    gateway 192.168.0.254
    bridge_ports eth0
    bridge_stp off
    bridge_fd 0

auto eth1
iface eth1 inet static
    address 192.168.1.2
    netmask 255.255.255.0

auto eth2
iface eth2 inet static
    address 192.168.2.2
    netmask 255.255.255.0
#Clusternetzwerk
```

Um das Neue Netzwerk zu aktivieren startet man die Server einfach durch. Man kann aber auch den Netzwerkdienst neustarten:

```
/etc/init.d/networking restart
```

Cluster generieren und Node hinzufügen:

```
pvecm create proxmoxcl01 -bindnet0_addr 192.168.1.0 -ring0_addr 192.168.1.1
```

```
-bindnet1_addr 192.168.2.0 -ring1_addr 192.168.2.1
```

Auf dem zweiten Host einloggen:

```
pvecm add 192.168.1.1 -ring0_addr 192.168.1.2 -ring1_addr 192.168.2.2
```

Cluster mit IPV6 generieren

Da hier keine öffentlichen (RADVD) adressen von nöten sind holt man sich einfach ein Subnet. Z.B. fddd:a4fb:635c:b9ef::/64. Von der Config ist zwischen IPV4 und IPV6 kein Unterschied. Lediglich mit den Multicast muss man ein wenig aufpassen. Unsere Netzwerkconfig würde dann so aussehen. **Netzwerkconfig pve01:**

```
auto lo
iface lo inet loopback

iface eth0 inet manual

auto vmbr0
iface vmbr0 inet static
    address 192.168.0.1
    netmask 255.255.255.0
    gateway 192.168.0.254
    bridge_ports eth0
    bridge_stp off
    bridge_fd 0

iface vmbr0 inet6 static
    address 2001:470:2a1c:561:eb5:7aff:fa2f:f7b2
    netmask 64
    gateway 2001:470:2a1c:561::8

auto eth1
iface eth1 inet6 static
    address fddd:a4fb:635c:b9ef::1
    netmask 64
#Clusternetzwerk
```

Netzwerkconfig pve02:

```
auto lo
iface lo inet loopback

iface eth0 inet manual

auto vmbr0
iface vmbr0 inet static
    address 192.168.0.2
    netmask 255.255.255.0
```

```
gateway 192.168.0.254
bridge_ports eth0
bridge_stp off
bridge_fd 0

iface vmbr0 inet6 static
    address 2001:470:2a1c:561:ec4:7aaf:fe21:f7a1
    netmask 64
    gateway 2001:470:2a1c:561::8

auto eth1
iface eth1 inet6 static
    address fddd:a4fb:635c:b9ef::2
    netmask 64
#Clusternetzwerk
```

Um das Neue Netzwerk zu aktivieren startet man die Server einfach durch. Man kann aber auch den Netzwerkdienst neustarten:

```
/etc/init.d/networking restart
```

Cluster generieren und Node hinzufügen:

```
pvecm create osit-cl01 -bindnet0_addr fddd:a4fb:635c:b9ef::1 -ring0_addr
fddd:a4fb:635c:b9ef::1
pvecm add fddd:a4fb:635c:b9ef::1 -ring0_addr fddd:a4fb:635c:b9ef::2
```

Unser Cluster sieht nun so aus:

```
Quorum information
-----
Date:                Fri Apr 14 13:54:29 2017
Quorum provider:     corosync_votequorum
Nodes:               2
Node ID:              0x00000001
Ring ID:              1/344
Quorate:              Yes

Votequorum information
-----
Expected votes:      2
Highest expected:    2
Total votes:         2
Quorum:              2
Flags:                Quorate

Membership information
-----
    Nodeid      Votes Name
0x00000001      1 fddd:a4fb:635c:b9ef::1 (local)
```

0x00000002

1 fddd:a4fb:635c:b9ef::2

From:

<https://deepdoc.at/dokuwiki/> - DEEPDOC.AT - enjoy your brain

Permanent link:

https://deepdoc.at/dokuwiki/doku.php?id=virtualisierung:proxmox_kvm_und_lxc:einrichtung_eins_standard_proxmox4_clusters&rev=1492171079

Last update: 2017/04/14 13:57

