2. Lab: Ceph

Modul Cloud Computing (cloud)

Simon Wächter, Jilin Elavathingal

2019

Inhalt

[1 Installation 2](#_Toc22402057)

[1.1 Hinweis 2](#_Toc22402058)

[1.2 Installationsvorbereitungen (Preflight Guide) und eigentliche Installation 0](#_Toc22402059)

[1.3 Hinzufügen des Ceph Clusters in Proxmox 0](#_Toc22402060)

[1.4 Fazit 0](#_Toc22402061)

[2 Fragen 1](#_Toc22402062)

[2.1 Leistungsdaten der VM Storage 1](#_Toc22402063)

[2.2 Zustand des Ceph-Clusters 1](#_Toc22402064)

# Installation

## Hinweis

Wir beide haben mit dem Lab am Freitag angefangen und unsere Schritte dokumentier, sind aber nicht fertig geworden. Herr Wächter wollte dann mit diesen Schritten zuhause in seinem Homelab die Übung abschliessen. Dabei hätten sich folgende Änderungen ergeben:

* IP und DNS sind wieder unterschiedlich
* Da auf der ersten Festplatte kein Platz für eine XFS Partition mehr vorhanden ist, wird eine zweite Festplatte via /dev/sdb eingehängt

Bei der weiteren Bearbeitung des Labs sind wir aber auf diverse Fehler gestossen. Nach dem dritten Anlauf (3 Mal System neu bootstrappen mit Ceph + 3 Mal Cephaufbau durchspielen = Circa 4h Zeitaufwand am Samstag) mussten wir leider aufgeben. Probleme und Gründe:

* Der initiale Versuch, mit Proxmox 5.4/Debian Stretch und Ceph Nautilus zu installieren, scheiterte in der Erstellung der OSD: ceph-deploy missing ceph-volume: <https://www.reddit.com/r/ceph/comments/bx7rzs/cephdeploy_missing_cephvolume/>. Generell scheint es, dass die Wahl von RHEL/CentOS für das Betreiben eines Ceph Clusters die stabilere Wahl ist (Zumal CentOS ja faktisch 1:1 kompatibel zur jeweiligen RHEL Version ist). Zumal es einen Mirror gibt (Sowas lässt aufhorchen): <https://croit.io/2019/07/07/2019-07-07-debian-mirror>
* Der zweite Versuch scheitert bei der Erstellung des Managers, da es im Verzeichnis /var/lib/.. zu Berechtigungsproblemen kam
* Der dritte Durchlauf mit Ceph Luminous scheiterte an Abhängigkeitsproblemen von apt-get bei der Installation von ceph und ceph-deploy an einem Mismatch von den vom Betreibssystem angebotetenen versus benötigten Versionen.

Möchte man mehr Zeit investieren, wäre das Lab wahrscheinlich mit folgenden Alternativen zu lösen:

* Man setzt rasch 3 RHEL/CentOS VMs auf und installiert Ceph in diesen Maschinen. Jedoch wird sicherlich die Disk-IO Performance des Ceph Storages leiden, da dieser virtualisiert wird
* Man wechselt auf Proxmox 6.0+ und hofft auf eine bessere Stabilität, da dort wahrscheinlich auf Debian 10 gesetzt wird

Damit die Zeitinvestitionen in das Lab aber sichtbar sind und die Abgabe hoffentlich zählt, möchten wir nachfolgend unsere verwendeten Befehle beschreiben und die Theoriefragen theoretisch beantworten (Hoffentlich geht dies in Ordnung).

## Installationsvorbereitungen (Preflight Guide) und eigentliche Installation

Die Preflight Installation wird auf allen drei Nodes ausgeführt:

|  |
| --- |
| # Packet mit FHNW CA installieren  wget --no-check-certificate http://srvtol01.cyberlab.fhnw.ch/repos/cyberlab/pool/main/c/cyberlab-trust/cyberlab-trust\_20181019-1\_all.deb  apt-get install -y ./cyberlab-trust\_20181019-1\_all.deb  # Proxmox PVE Enterprise Package Repository deaktivieren  echo "#deb https://enterprise.proxmox.com/debian/pve stretch pve-enterprise" > /etc/apt/sources.list.d/pve-enterprise.list  # Ceph Package Repository registrieren  wget -q -O- 'https://download.ceph.com/keys/release.asc' | apt-key add -  echo deb https://download.ceph.com/debian-nautilus/ stretch main | tee /etc/apt/sources.list.d/ceph.list  # Alle Abhängigkeiten installieren  apt-get update -y  apt-get install -y sudo ntp ntpdate ntp-doc python python-pip parted uuid-runtime openssh-server ceph ceph-deploy 🡪 Hier gab es im 2. Durchlauf Abhängigkeitsprobleme bei der Installation von ceph/ceph-deploy mit den Paketen des Betriebssystems  # Systembenutzer erstellen  useradd -d /home/cephuser -m cephuser  passwd cephuser  # Passwordless sudo erlauben  echo "cephuser ALL = (root) NOPASSWD:ALL" | tee /etc/sudoers.d/cephuser  chmod 0440 /etc/sudoers.d/cephuser  # Allfälliges requiretty deaktivieren (Eher auf RHEL/CentOS)  --> Defaults:ceph !requiretty  sed -i s'/Defaults requiretty/#Defaults requiretty'/g /etc/sudoers  # NTP synchronisieren  systemctl stop ntp  ntpdate time.google.com  hwclock --systohc  systemctl enable ntp  systemctl start ntp  # OK: Die Lab Images verwenden keine Firewall, von dem her haben wir das nicht getestet  # Firewallregeln anpassen (Eventuell optional, wenn kein UFW vorhanden)  ufw enable  ufw allow 22/tcp  ufw allow 80/tcp  ufw allow 2003/tcp  ufw allow 4505:4506/tcp  ufw allow 6789/tcp  ufw allow 6800:7300/tcp  # SSH Konfiguration erstellen 🡪 Ich kann hier keine nicht DNS fähigen SSH Kurznamen wie beispielsweise node1 für pve1.swaechter.ch wählen - Idee?  mkdir /home/cephuser/.ssh  bash -c 'cat > /home/cephuser/.ssh/config <<EOF  Host pve1.swaechter.ch  Hostname pve1.swaechter.ch  User cephuser  Host pve2.swaechter.ch  Hostname pve2.swaechter.ch  User cephuser  Host pve3.swaechter.ch  Hostname pve3.swaechter.ch  User cephuser  EOF'  chown -R cephuser:cephuser /home/cephuser/.ssh  chmod 700 /home/cephuser/.ssh  # Festplatte für den OSD Speicher vorbereiten  fdisk -l /dev/sdb  parted -s /dev/sdb mklabel gpt mkpart primary xfs 0% 100%  mkfs.xfs -f /dev/sdb |

Danach kann von der Adminnode aus die Installation vorangetrieben werden:

|  |
| --- |
| # Benutzer wechseln  su - cephuser  # SSH Key ohne Passphrase erstellen  ssh-keygen  # SSH Key auf andere Nodes verteilen  ssh-copy-id cephuser@pve2.swaechter.ch  ssh-copy-id cephuser@pve3.swaechter.ch  # Verzeichnis erstellen  mkdir my-cluster  cd my-cluster  # Ceph Cluster Konfiguration + lokale Monitor Konfiguration erstellen  ceph-deploy new pve1.swaechter.ch  # Ceph auf allen Nodes erstellen  ceph-deploy install pve1.swaechter.ch pve2.swaechter.ch pve3.swaechter.ch  # Initialen Monitor starten + Keys sharen  ceph-deploy mon create-initial  ceph-deploy gatherkeys pve1.swaechter.ch  # Konfigurationsdatei und Admin Keys auf alle Nodes kopieren  ceph-deploy admin pve1.swaechter.ch pve2.swaechter.ch pve3.swaechter.ch  # Manager erstellen und starten 🡪 Hier gab es im 3. Durchlauf diverse Accesserrors in /var/lib/…  ceph-deploy mgr create pve1.swaechter.ch  # Disks prüfen  ceph-deploy disk list pve1.swaechter.ch pve2.swaechter.ch pve3.swaechter.ch  # OSDs auf allen Nodes erstellen 🡪 Hier sind im 1. Durchlauf die ceph-volume command not found Fehler entstanden  ceph-deploy osd create --data /dev/sdb pve1.swaechter.ch Ab hier alles theoretisch mit den Guides!  ceph-deploy osd create --data /dev/sdb pve2.swaechter.ch  ceph-deploy osd create --data /dev/sdb pve3.swaechter.ch  # Gesundheit und Status prüfen  ceph health  ceph status  # Optional für Nodes  ssh <IP> sudo ceph -s  # Metadata Server auf allen Nodes erstellen und starten  ceph-deploy mds create pve1.swaechter.ch pve2.swaechter.ch pve3.swaechter.ch  # Weitere Monitore erstellen, um High Availability zu erreichen  ceph-deploy mon add pve2.swaechter.ch  ceph-deploy mon add pve3.swaechter.ch  # Weitere Manager erstellen, um High Availability zu erreichen  ceph-deploy mgr create pve2.swaechter.ch  ceph-deploy mgr create pve3.swaechter.ch |

Hinweis: Jetzt sind alle 3 Nodes à la HCI gleichwertig. Bei mehr als 3 Nodes würde man natürlich nicht auf jeder Node einen Monitor, Manager, MDS und OSD erstellen 🡪 Overkill + gefährlich, da der Quorumnetzwerkverkehr Überhand nehmen kann

## Hinzufügen des Ceph Clusters in Proxmox

Das Hinzufügen des Ceph Storages in Proxmox wurde nicht getestet, aber wird in diesem Video recht gut beschrieben (Keine grosse Sache): <https://www.youtube.com/watch?v=a4swML-TNXs>

Ab Minute 13:50 wird eine interessante Aussage gemacht, welche uns auch so vorgeschwebt ist, wenn man sich mal etwas über Blockspeicher vs. Objektspeicher im Rahmen als VM Speicher einliest: External ceph cluster storage You don’t want to do hyper converged [virtualization] … I don’t know which idiot came up with that 🡪 Performance isn’t there (Klar hier handelt es sich um eine Aussage ohne unterlegte Fakten, aber das war auch unser Eindruck)

## Fazit

Es ist schade, dass wir das Lab nicht umsetzen konnten und aufgeben mussten. Wahrscheinlich wäre es einfacher gewesen, via UI-Setup aus Proxmox heraus Ceph installieren zu lassen oder die CLI Tools von Proxmox zu verwenden (Oder halt das Skriptkiddie-Skript zu verwenden…).

# Fragen

## Leistungsdaten der VM Storage

Auszuführende Befehle in der VM:

|  |
| --- |
| sudo apt-get install bonnie++  sudo bonnie -u root |

Da wir dies nicht ausführen konnten, sind folgende Annahmen zu tätigen:

* Lokaler Storage ist schneller als ein entfernter Block- oder Objectstorage, sofern Disk-IO nicht der limitierende Faktor ist
* Ein Blockspeicher auf Basis von NFS wird schneller sein als ein «gleichwertiger» Objektspeicher, da ein Objektspeicher über eine grössere Abstraktion verfügt. Doch verfügt ein Blockspeicher auch über weniger Features (Kein Objektaufbau, kein/wenig flexibleres Abspeichern von Metadaten etc. 🡪 Blockspeicher ist halt «dumm»)
* Sobald aber (lokaler) Disk-IO zu einem limitierenden Faktor wird, sind natürlich Block- und Objektspeicher wieder interessant. Heisst: Ab wann ist es schneller, eine Datenmenge X auf mehreren entfernten Systemen abzuholen, als sie ausgelastet lokal zu lesen & auf eine limitierende Disk zu treffen
* Seitengedanke: Blockspeicher weissen in der Regel nicht die verteilten und HA Features von abstrahierten Objektspeicher auf:
  + Ein verteilter und HA fähiger NFS Storage scheint nicht sehr verbreitet zu sein. Zwar gibt es ein paar Produkte wie LzardFS, doch scheint es sich um Nischenprodukte zu handeln (Heisst es gibt keine sehr dominanten Produkte wie Ceph oder GlusterFS für verteilte Objektspeicher)
  + iSCSI wird in vielen Fällen auch in einer 1:1 Beziehung zwsichen Initiator und Target Node verwendet. Zwar gibt es Produkte, welche eine n:1 Beziehung ermöglichen, doch bringen diese einen erhöten Konfigurationsaufwand + schränken unter Umständen die Verwendung ein
* Annahme: Der Vergleich zwischen lokalen Speicher/NFS Speicher und Objektspeicher wird zeigen, dass Objektspeicher zu langsam ist (Sozusagen ein Untermauern der Aussage aus dem Video zur Verwendung von Ceph als VM Store)

## Zustand des Ceph-Clusters

Da wir dies nicht ausführen konnten, sind folgende Annahmen zu tätigen:

* Eine virtuelle Maschine besteht in der Regel aus mehreren Dateien (VM Datei, weitere Datei für jede HDD etc.)
* Entweder ist jede physikalische Datei ein logisches Objekt im Ceph Speicher oder aber Proxmox splittet VMs in mehrere Objekte auf. (Denken wir eher nicht, da wir dann wieder auf das Problem des Random Read antreffen)