

# PROXMOXVE

Konzepte und Technologien der freien Virtualisierungssoftware



THOMAS  
KRENN®



THOMAS  
KRENN®  
IT's people business

# Inhaltsverzeichnis

<b>1. Einleitung</b>	3
<b>2. Konzepte</b>	3
Lizenzierung und Support-Modelle	5
Community Support	6
Entwicklungsprozess und Versionierung	
<b>3. Hardware-Virtualisierung</b>	6
Paravirtualisierte Treiber	7
Live-Migration	
Linked Clones	
Cloud-Init	
GPU-Unterstützung & GPU-Virtualisierung	8
<b>4. Container-Virtualisierung</b>	8
Proxmox VE Container Toolkit	
Container-Templates	9
Container und Storage	
<b>5. Das Proxmox VE Netzwerk</b>	9
Bonding	10
Virtual LAN	
<b>6. Dateisysteme und Storage</b>	11
Konzept und Terminologie	
PVE Type	
Content Types	
Image-Formate	13
Storage-Replication	
<b>7. Hochverfügbarkeit</b>	13
<b>8. Proxmox VE administrieren</b>	14
Grafisches User-Interface	
Kommandozeilen-Tools	15
API	16
<b>9. Tipps zur Hardware</b>	17
CPU	
Hauptspeicher	
Netzwerk	
RAID-Controller	18
Proxmox VE & Ceph HCI	
<b>10. Fazit</b>	19

# 1. Einleitung

## Proxmox VE - Konzepte und Technologien der freien Virtualisierungssoftware

Die Virtualisierungsplattform Proxmox VE hat sich in den letzten Jahren vom Geheimtipp zu einer ernst zu nehmenden Alternative auch im Unternehmenseinsatz entwickelt. Bei Thomas-Krenn steigt die Nachfrage nach Systemen, die für Proxmox optimiert sind, ständig an. Proxmox VE ist als reine Open Source Umgebung günstiger im Betrieb als proprietäre Lösungen und dabei sehr flexibel hinsichtlich Verwendungszweck und

Hardware-Anforderungen. Die Möglichkeiten reichen dabei von einfachen Hosts für virtuelle Maschinen oder Container über ausfallsichere Systeme bis hin zu hyperkonvergenten Clustern und virtualisierten Speichersystemen für Software Defined Storage.

Dieses E-Book stellt die Konzepte und Technologien von Proxmox VE vor und geht dabei auf die Vor- und Nachteile der Software ein.

## 2. Konzepte

Proxmox VE ist eine Umgebung zur Einrichtung und zum Betrieb virtueller Umgebungen. Dabei wird sowohl Hypervisor-basierte als auch Container-basierte Virtualisierung unterstützt. Mittels Proxmox VE erfolgt die Konfiguration und Administration der Virtualisierungshosts, der virtuellen Netzwerke und des gemeinsamen Storage. Es lässt sich auch auf einzelnen Servern betreiben, seine Stärke sind jedoch Cluster aus bis zu 32 Nodes, die als ein einziges System betrachtet werden und ohne weiteres Tausende virtuelle Maschinen beherbergen können.

Die eigentliche Virtualisierungs-, Netzwerk- und Storage-Technologie wird nicht von den Proxmox-Machern entwickelt, sondern ist entweder bereits Bestandteil von Standard-Linux-Distributionen oder kommt von dritten, ausgereiften Open Source Projekten und wird in die Umgebung integriert. Proxmox VE basiert dabei auf Debian GNU/Linux. Streng genommen stellt es eine eigene, von Debian abgeleitete Linux-Distribution dar, die zahlreiche, miteinander integrierte Werkzeuge zur Verwaltung der Linux-eigenen Virtualisierungskomponenten enthält. Viele davon sind in eine Web-Oberfläche integriert, andere kommen in Form typischer Kommandozeilen-Tools, die im Terminal ausgeführt werden. Gute Linux-Kenntnisse sind also zumindest bei der Konzeption und Inbetriebnahme einer Proxmox-Umgebung unerlässlich.

Proxmox VE besitzt eine eigene Installationsroutine

und wird in der Regel komplett von einem Installationsmedium oder über das Netzwerk installiert. Zusätzlich lässt sich auch eine bereits vorhandene Debian-Installation in Proxmox VE verwandeln, indem man die entsprechenden Pakete nachinstalliert. Auch für Test-Installationen sollte nach Möglichkeit ein kompletter Rechner einschließlich komplett für Proxmox VE nutzbarer Speichermedien zur Verfügung stehen. Die bei Linux oft üblichen Parallel-Installationen neben anderen Systemen unterstützt die Installationsroutine nicht. Charakteristisch für Proxmox VE ist, dass es mehrere Technologien integriert, die sich nahtlos ergänzen: Von Hypervisor über Container über eine Vielzahl von Storage-Technologien (z.B. ZFS oder Ceph) bis hin zu Bridges oder VLANs für die Netzwerk-Konfiguration. Seit Version 8.1 kann die Konfiguration via Software-Defined Networking auch zentral auf der Datacenter-Ebene erfolgen.

Die Anwender haben somit die Möglichkeit, die für ihre Aufgaben optimalen Werkzeuge zu wählen.

Der Hypervisor in Proxmox VE ist KVM ("Kernel Based Virtual Machine"), ergänzt durch QEMU ("Quick Emulator") zur Emulation (Abbildung) der Hardware eines Computers. Beides sind Bestandteile des Linux-Kernels bzw. der Systemwerkzeuge jeder Linux-Distribution. Für die Container-Virtualisierung kommt LXC ("Linux Containers") zum Einsatz, auch diese sind im Standard-Linux enthalten. Abbildung 1 zeigt den prinzipiellen Aufbau.

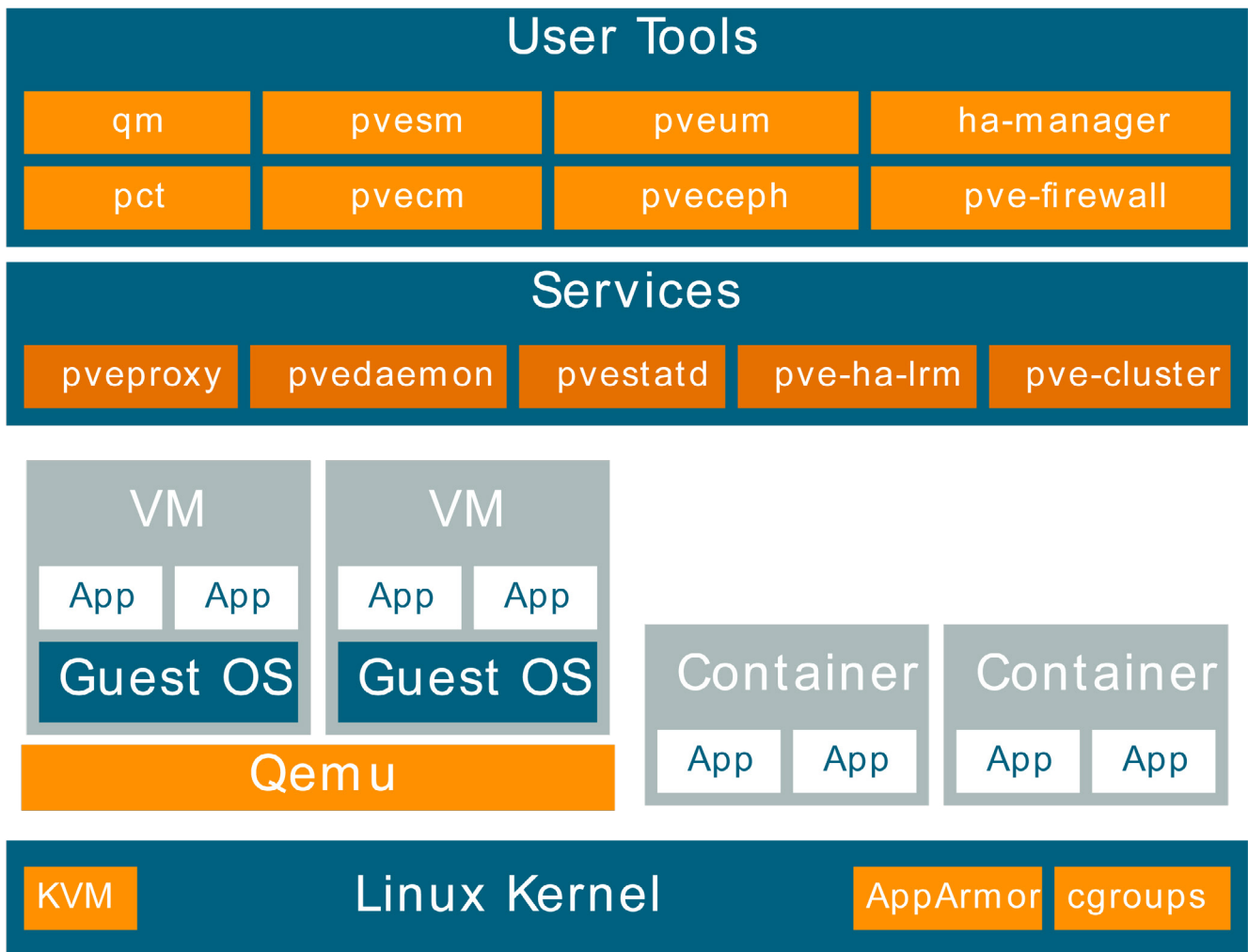


Abbildung 1: Architektur von Proxmox VE, Quelle: Proxmox Server Solutions GmbH

KVM/QEMU und LXC sind ausgereifte Technologien, die seit mehr als einem Jahrzehnt entwickelt werden, unter anderem von IBM, HP, Red Hat und SUSE Linux. Das Bundesamt für Informationssicherheit (BSI) hat 2017 in einer Sicherheitsanalyse festgestellt, dass KVM/QEMU "dazu geeignet ist, eine technisch ausgereifte und sichere Virtualisierungsumgebung zu realisieren".

Im Storage-Bereich ist bei Proxmox VE vor allem die Unterstützung des Dateisystems ZFS hervorzuheben. Es ist für die effiziente, ausfallsichere Speicherung praktisch unbegrenzter Datenmengen konzipiert und wird in zahlreichen kommerziellen Produkten für Software Defined Storage verwendet, etwa dem von Thomas-Krenn vertriebenen JovianDSS von Open-E oder den Open Storage Systemen (OSS).. Zusätzlich unterstützt Proxmox VE auch Ceph, die verteilte Plattform für

Object Storage, die unter anderem von Red Hat, SUSE, IBM, Intel und Fujitsu mitentwickelt wird. Die Administrationsoberfläche ist einfach gehalten, für den Zugriff genügt ein Webbrowser auf einen beliebigen Host im Cluster. Das User Interface ist responsiv gestaltet und deshalb auch mit Smartphones oder Tablets benutzbar. Für Android gibt es auch eine offizielle Mobile-App für Zugriff und Management eines Proxmox VE Systems.

Proxmox VE basiert auf der Debian GNU/Linux-Distribution, jedoch setzt es einen eigenen Kernel mit zusätzlichen Treibern ein, etwa für das Filesystem ZFS und zur Unterstützung aktueller Netzwerkarten, die in den Kernen von Standard-Distributionen fehlen. Dennoch ist der "Unterbau" von Proxmox VE ein normales Debian-System, das mit den gängigen Linux-Werkzeugen administriert werden kann.

# Lizenzierung und Support-Modelle

Wie jede Open Source Software ist Proxmox VE zwar für private und kommerzielle Nutzung kostenlos, untersteht aber dennoch einer Lizenz. Das heißt, die Verwendung ist an bestimmte Bedingungen geknüpft. Die Proxmox-Entwickler haben sich für die Affero General Public License (AGPL) (<https://www.gnu.org/licenses/agpl-3.0.en.html>) entschieden.

Etwas vereinfacht bedeutet das Folgendes: Anwender können die Software für beliebige Zwecke nutzen und Proxmox-Systeme beliebig vielen Usern zu Verfügung stellen. Auch eigene Anpassungen der Software sind erlaubt. Wer jedoch die Software verändert oder erweitert, muss den Quellcode der Änderungen und Erweiterungen jedem zugänglich machen, der Zugriff auf die Systeme hat. Und zwar unter den gleichen Bedingungen wie die Original-Software, also mit der AGPL lizenziert. Damit soll verhindert werden, dass Closed-Source-Versionen der Software entstehen.

Da Proxmox VE kostenlos erhältlich ist, ist jeder Anwenderselbst für die Wartung verantwortlich oder er überträgt sie einem Dienstleister, idealerweise einem Systemhaus mit sehr gutem Linux-Know-how. Alternativ bietet die Proxmox Server Solutions GmbH in Wien mehrere Support-Modelle an, deren Preis sich nach der Anzahl der CPUs richtet, auf denen Proxmox verwendet wird. Im Vergleich zum Enterprise-Support großer Software-Unternehmen sind die Support-Modelle ausgesprochen günstig. So ist im teuersten Paket "Premium Support" pro

Jahr und CPU eine unbegrenzte Anzahl Support-Tickets enthalten, bei einer garantierten Antwortzeit von zwei Stunden bei kritischen Problemen. Die zwei günstigeren Pakete "Standard" und "Basic" unterscheiden sich in der Anzahl der enthaltenen Support-Tickets und der garantierten Antwort-Zeit ("Standard": 10 Tickets pro Jahr bei vier Stunden, "Basic": Drei Tickets pro Jahr bei Antwort innerhalb eines Arbeitstages). Ein Sonderfall ist der so genannte Community Plan. Hier sind keine Tickets enthalten, der Kunde erhält aber Zugriff auf das Enterprise Repository, was sicherstellt, dass nur die vom Hersteller getesteten und als fehlerfrei anerkannten Patches eingespielt werden.

Alle Support-Pläne haben zwei wesentliche Einschränkungen. Zum einen gibt es keine Telefon-Hotline. Anfragen müssen immer über das Ticket-System im Customer Portal gestellt werden. Zweitens wird Support nur während der Geschäftszeiten von 7 Uhr bis 17 Uhr an österreichischen Arbeitstagen geleistet. Die oben genannten Antwortzeiten werden ebenfalls nur innerhalb dieser Geschäftszeiten garantiert. Einen echten 24/7-Support bietet die Proxmox Server Solutions GmbH nicht an. Andererseits erfolgt der Support immer direkt durch qualifizierte Mitarbeiter der Proxmox Server und Solutions GmbH. Ein ausgelagertes Call Center für First Level Support gibt es nicht.

# Community Support

Den Proxmox-Entwicklern ist es gelungen, eine funktionierende, aktive Community rund um Proxmox VE aufzubauen. Sie betreiben ein sehr aktives öffentliches Support-Forum (<https://forum.proxmox.com/>) mit rund 150.000 Nutzern und

über 590.000 Einträgen. Für konkrete technische Probleme findet sich hier oft eine Lösung. Daneben existiert ein Wiki ([https://pve.proxmox.com/wiki/Main\\_Page](https://pve.proxmox.com/wiki/Main_Page)) mit allgemeiner gehaltenen technischen Artikeln und Howtos.

## Entwicklungsprozess und Versionierung

Neue Proxmox-Versionen erscheinen etwa zwei- bis dreimal im Jahr. Die Versionsbezeichnungen folgen einem einfachen Muster aus Major- und Minor-Nummerierung. Neue Major-Versionen erscheinen immer dann, wenn im Zuge des Debian-Entwicklungsprozesses eine neue Debian-Version

zu "Stable" erklärt wird. Wichtige geplante Features kommender Versionen geben die Entwickler in einer kurzen Roadmap vorher bekannt. (<https://pve.proxmox.com/wiki/Roadmap>)

## 3. Hardware-Virtualisierung

Im Vergleich zu VMware vSphere oder Microsoft Hyper-V ist die Proxmox-Virtualisierung mittels KVM/QEMU schlanker und modularer aufgebaut, aber ähnlich leistungsfähig. Streng genommen ist nämlich KVM kein Hypervisor, sondern lediglich ein eher kleines Kernel-Modul, das den Linux-Kernel des Hosts selbst zum Hypervisor macht, sobald es geladen wird. QEMU übernimmt die

Abbildung der realen Hardware des Hosts auf die der virtuellen Maschine. Dieses Prinzip der Hardware-Virtualisierung erlaubt es theoretisch, alle Betriebssysteme ohne Modifikationen für virtuelle Maschinen zu verwenden, die es für die x86-Architektur gibt. Nur bei sehr alter Software und 32-Bit-Systemen treten gelegentlich Probleme auf.

Hier eine kleine Auswahl der von Proxmox VE unterstützten Gast-Systeme:

- Microsoft Windows Server 2022, 2019, 2016, 2012, 2008, 2003
- Microsoft Windows 11, 10, 7
- Windows XP
- Red Hat Enterprise Linux von Version 4 bis 7
- Rocky Linux in aktueller Version
- Ubuntu
- FreeBSD, OpenBSD und NetBSD in fast allen Versionen von ca. 2006 bis heute

QEMU bietet außerdem die Möglichkeit, auch Systeme auszuführen, die eine andere Hardware-Architektur als x86 haben. Dies läuft dann ohne KVM ab und ist entsprechend langsamer. Proxmox

VE bietet jedoch keine Unterstützung für die Installation und Administration solcher Systeme an.



## Paravirtualisierte Treiber

Im Gegensatz zu reiner Hardware-Virtualisierung ist Paravirtualisierung ein Konzept, bei dem das Gast-Betriebssystem direkt mit dem Hypervisor kommuniziert, also "weiß", dass es in einer virtualisierten Umgebung läuft.

Das bringt Vorteile bei der Performance, aber auch besseres Management und Monitoring der Gäste, da der Hypervisor deren Zustand genau kennt. Proxmox VE unterstützt die teilweise Paravirtualisierung durch spezielle Treiber nach

dem virtio-Standard für Netzwerk-Interfaces, Block Devices und das so genannte Ballooning. Letzteres erlaubt es, virtuellen Maschinen Hauptspeicher dynamisch nach Bedarf zuzuweisen und so die RAM-Ressourcen des Hosts effektiv zu nutzen.

Die virtio-Treiber sollten nach Möglichkeit immer installiert und aktiviert werden. Deren Installation für Windows wird in der Proxmox-Dokumentation ausführlich beschrieben. (<https://pve.proxmox.com/pve-docs/pve-admin-guide.html>)

## Live-Migration

Proxmox VE erlaubt die Live-Migration, bei Proxmox auch Online-Migration genannt. Das bedeutet, virtuelle Maschinen lassen sich unter bestimmten Voraussetzungen von einem ProxmoxVE-Host auf den anderen verschieben ohne dass der Betrieb

des Gast-Systems unterbrochen wird. Eine Live-Migration ist auch bei lokalem Speicher möglich, allerdings muss hierbei die komplette virtuelle Maschine inklusiver allen virtuellen Festplatten über das Netzwerk in voller Größe kopiert werden.

## Linked Clones

Cloning von existierenden virtuellen Maschinen ist ein beliebtes Verfahren, neue, identisch konfigurierte virtuelle Maschinen mit wenig Aufwand zu erstellen. Neben so genannten Full Clones erlaubt Proxmox VE auch Linked Clones. Bei diesen werden die unmodifizierten Daten vom Ausgangsimage gelesen, Modifikationen jedoch an eine neue Stelle im Storage geschrieben. Dieses Verfahren, auch Copy-on-Write genannt, spart

also viel Speicherplatz. Im Gegensatz zum Full Cloning ist der Linked Clone jedoch während seiner gesamten Lebenszeit abhängig vom Original, letzteres darf als nicht gelöscht oder modifiziert werden. Deshalb muss bei Proxmox VE das Original ein so genanntes Template sein. Ein Template ist eine schreibgeschützte Vorlage, die sich jederzeit aus einem Image generieren lässt.

## Cloud-Init

Bei Cloud-Anbietern wie AWS oder Microsoft Azure ist Cloud-Init (<https://cloud-init.io/>) der herstellerübergreifende Standard, um neu anzulegende virtuelle Maschinen automatisiert mit individuellen Konfigurationen zu versehen. Diese Aufgabe kann auch ein Proxmox-Host erfüllen.

Dazu werden Templates und Clones benutzt. Die individuellen Daten für neue Maschinen liest der Clone von einem virtuellen CD-Laufwerk ein. Auf diese Weise können Nutzer angelegt, das Netzwerk konfiguriert und SSH-Schlüssel eingerichtet werden.

## GPU-Unterstützung & GPU-Virtualisierung

GPU Passthrough, also das direkte Durchreichen des Grafik-Outputs, wird unterstützt, erfordert aber eine Remote-Desktop-Software wie etwa VNC oder RDP. Wird die GPU lediglich benutzt, um die Rechenleistung des Systems zu erhöhen (GPU Computing etwa für KI-Anwendungen), ist diese Einschränkung nicht relevant.

Als Grafik-Devices innerhalb virtueller Maschinen lassen sich auch virtuelle GPUs nutzen, falls die Hardware das Feature bereitstellt. Die entsprechende Technik heißt bei Intel GVT-g und

bei NVIDIA vGPU. Damit lassen sich vorhandene physische Grafikkarten in mehrere virtuelle GPUs aufspalten. Bei Proxmox VE heißt das entsprechende Feature unabhängig von der Technologie "Mediated Device". Sowohl für Intel als auch für NVIDIA ist etwas mehr Konfigurationsarbeit zu leisten. Außerdem erfordert die Nutzung des Features eine kostenpflichtige Lizenz von NVIDIA.

## 4. Container-Virtualisierung

Container sind eine sehr leichtgewichtige Art der Virtualisierung, die bei Cloud-Anbietern und auch bei der Entwicklung moderner modularer Software wie Microservices äußerst beliebt ist. Der Erfolg etwa von Docker und darauf aufsetzender Frameworks wie Kubernetes sind der beste Beweis dafür. Proxmox VE verwendet jedoch keine Docker-Container, sondern LXC ("Linux Containers"). Im Gegensatz zu virtuellen Maschinen werden Container nativ auf dem Host-Betriebssystem eingesetzt. Dieses sorgt dafür, dass für jeden Container eine isolierte Umgebung zur Verfügung steht, sperrt also die Prozesse ein (Bei BSD-Unix-Systemen, in denen es das Konzept schon sehr lange gibt, heißen sie sogar Jails). Da dabei so gut wie kein Overhead durch

separate Gast-Kernel oder Hardware-Emulation entsteht, sind problemlos hunderte Container auf einem Host möglich. Die Isolation der Gastsysteme ist jedoch nicht so vollständig wie bei virtuellen Maschinen. Das heißt, Container-Virtualisierung ist immer ein größeres Sicherheitsrisiko als Hardware-Virtualisierung.

Container erlauben auch nur Gastsysteme vom gleichen Typ wie das Host-System, bei Proxmox VE also Linux. Jedoch sind sie nicht auf die jeweilige Linux-Distribution des Host-Systems beschränkt. Außer Debian-Varianten und Ubuntu können also auch Red Hat-, SUSE- oder Arch Linux-basierte Systeme betrieben werden.

## Proxmox VE Container Toolkit

Anders als Docker ist LXC lediglich eine Sammlung von Low-Level-Bibliotheken und sehr einfachen Verwaltungswerkzeugen, die für komplexere Konfigurationen schwer zu benutzen sind. Proxmox VE bringt deshalb das Proxmox Container Toolkit als Wrapper für die systemeigenen Werkzeuge mit. Ziel des Toolkits ist es, die Verwaltung von VMs und Containern so weit wie möglich anzugleichen.

Proxmox VE sorgt bei der Installation eines Containers automatisch dafür, dass das Gastsystem im Container an den Host angepasst wird und schreibt bzw. modifiziert dort die entsprechenden Konfigurationsdateien für Netzwerk, Init-Prozesse etc. Außerdem nutzt es Mechanismen wie AppArmor und Seccomp, um die Sicherheit zu erhöhen.



# Container-Templates

Proxmox VE bietet eine Reihe von Templates für den sofortigen Einsatz als Container-Gastsystem an. Neben der jeweils aktuellen Version sind meist

auch jeweils zwei bis vier ältere dabei. Für folgende Betriebssysteme gibt es so genannte System-Templates:

- Alpine Linux
- Arch Linux
- CentOS
- Debian
- Fedora
- Gentoo
- OpenSUSE
- Ubuntu
- Devuan
- AlmaLinux
- Rocky Linux

Daneben wird auch eine Reihe von Debian-Templates aus dem Turnkey-Linux-Projekt unterstützt. Dies sind Images, die für eine bestimmte Server-Anwendung vorkonfiguriert sind. So lassen sich

beispielsweise beliebte Open Source Anwendungen wie Nextcloud, Magento, Mediawiki oder Wordpress sofort in Betrieb nehmen.

<https://www.turnkeylinux.org/>

## Container und Storage

Container können die gesamte Palette der von Proxmox VE zur Verfügung gestellten Storage-Möglichkeiten nutzen. Auch unterschiedliche Storage-Arten innerhalb eines Gastsystems sind möglich. So können zum Beispiel Datenbank-

Applikationen von schnellen, verteilten Storage-Systemen profitieren. Gastsysteme im Container können über iSCSI angebotenen Storage genauso nutzen wie beispielsweise ZFS und sogar Ceph-Speicher.

## 5. Das Proxmox VE Netzwerk

Virtualisierungssysteme benötigen flexible und deshalb komplexe Konfigurationsmöglichkeiten für das Netzwerk. Denn sie müssen in unterschiedlichen Umgebungen einsetzbar sein, etwa bei Hostern, in der Unternehmens-IT oder bei der Software-Entwicklung. Dazu müssen sie in der Lage sein, bei Bedarf hohen internen und externen Traffic zu bewältigen. Dennoch soll das Aufsetzen neuer Maschinen nicht durch komplizierte Netzwerk-Konfigurationen behindert werden. Proxmox VE löst diesen Widerspruch, indem es ein relativ einfaches Netzwerk-Konzept als Vorgabe setzt, aber weitreichende Konfigurationsmöglichkeiten mittels GUI oder Kommandozeilen-Tools bietet.

Alle Gast-Systeme auf einem Host teilen sich in

der Default-Installation eine Bridge (Virtueller Switch) und verhalten sich so, als ob sie direkt mit dem physischen Netzwerk verbunden sind. Das heißt auch, dass jedes Gast-System ein Netzwerk-Interface mit eigener MAC-Adresse besitzt (Abbildung 2). Es ist jedoch möglich, mehrere Bridges auf einem Host anzulegen und so virtuelle Maschinen bestimmten Netzwerk-Segmenten zuzuordnen. Falls das Netzwerk keine virtuellen MAC-Adressen erlaubt, lässt sich der Traffic über die Bridge auch routen, so dass das IP-Layer für die Verteilung der Pakete verantwortlich ist und MAC-Adressen vor dem externen Netzwerk verborgen bleiben.

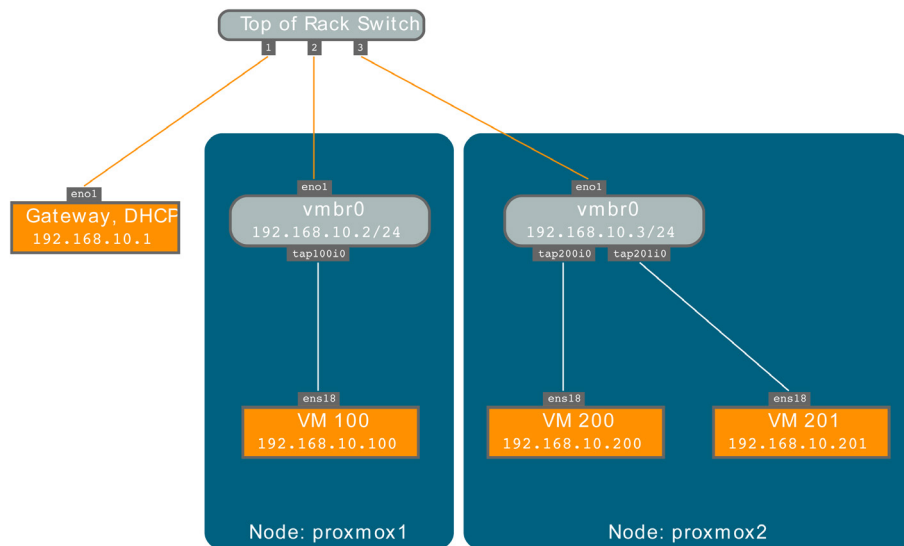


Abbildung 2: Zwei Proxmox-Nodes mit Netzwerk im Bridge-Modus, Quelle: Proxmox Server Solutions GmbH

Im einfachsten Fall, bei nur einem Host und einem Netzwerk-Interface, läuft der Traffic der virtuellen Maschinen und der des Hosts selbst über ein und dasselbe physische Interface. In der Regel führt das unter Produktionsbedingungen zu Problemen. Für Cluster aus mehreren Hosts sollte auf jeden Fall der Cluster-Traffic vom Host-Traffic separiert werden,

was ebenfalls innerhalb des Bridge-Konzepts leicht möglich ist. Proxmox VE bietet darüber hinaus viele Optionen, die Netzwerk-Architektur an individuelle Anforderungen anzupassen. Der Proxmox VE Administration Guide und das Proxmox-Wiki liefern dazu zahlreiche Konfigurationsbeispiele.

## Bonding

Bonding, auch Link Aggregation genannt, sorgt dafür, dass sich mehrere physische Netzwerkgeräte dem Betriebssystem gegenüber zusammen als ein einziges präsentieren. Damit kann es im einfachsten Fall dazu dienen, die Bandbreite zu erhöhen, aber auch für Ausfallsicherheit sorgen oder beides

zusammen. Proxmox VE erlaubt zahlreiche Bonding-Methoden, die effektivsten erfordern Switches mit den entsprechenden Fähigkeiten, andere, wie etwa Adaptive Transmit Load Balancing, arbeiten mit jedem Switch zusammen.

## Virtual LAN

Virtual LANs (VLANs) sind eine beliebte Methode, Netzwerk-Traffic zu organisieren und zu isolieren. Innerhalb eines physischen Netzwerks erhalten alle Pakete ein Tag, das sie einem virtuellen LAN zuordnet. Proxmox VE unterstützt VLANs nach dem Standard IEEE 802.1q, sowohl für Gastmaschinen als auch für den Host selbst. Der einfachste Weg, die "VLAN Awareness"-Methode, kommt ohne Konfiguration des Gast-Systems aus, da

die virtuelle Netzwerkkarte und der Switch des Hosts automatisch entsprechend konfiguriert werden. Weitere Möglichkeiten sind VLANs, die vollständig innerhalb des Gasts realisiert sind, um beispielsweise bei einem komplexen Server den Traffic von Anwendungen zu trennen. Ein anderer Ansatz ist es, für jedes VLAN ein eigenes virtuelles Interface mit einer dazugehörigen Bridge auf dem Hostsystem zu erstellen.

## 6. Dateisysteme und Storage

Die Auswahl des passenden Storage-Modells ist einer der wichtigsten Schritte beim Aufsetzen einer Virtualisierungsumgebung. Im laufenden Betrieb lässt sie sich nur mit viel Aufwand korrigieren und hängt auch davon ab, wie leistungsfähig die zur Verfügung stehende Hardware ist. Einige Tipps zur Hardware-Auswahl für Proxmox VE, vor allem in Hinblick auf Storage, finden Sie im letzten Abschnitt dieses E-Books.

Eine der größten Stärken von Proxmox VE ist seine Unterstützung mehrerer leistungsfähiger Storage-Technologien aus der Open Source Welt. Jeder Anwender wird also mit ziemlicher Sicherheit ein geeignetes Storage-Modell finden können. Dazu ist es aber im ersten Schritt nötig, über die Storage-Konzepte von Proxmox VE Bescheid zu wissen.

Grundsätzlich muss man beim Ort der Speichermedien zwischen internem und externem Storage wie NFS oder SAN unterscheiden und

zusätzlich bei einem Cluster aus mehreren Nodes auch nach der Anbindung zwischen lokalem und Shared Storage. Diese beiden Unterscheidungen sind wichtig, denn beispielsweise kann bei einem Cluster interner Storage mehrerer Nodes zu Shared Storage verbunden werden, den virtuelle Maschinen aller Nodes gemeinsam nutzen (Hyperkonvergenz). Andererseits können externe Storage-Systeme auch nur lokal an einem Node angebunden sein. Die dritte Unterscheidung ist die zwischen Block- und File-Storage.

Für das Storage von virtuellen Maschinen auf Clustern bestehend aus mehreren Nodes sollte wann immer möglich einem Shared-Storage-Konzept der Vorzug gegeben werden. Es erhöht die Flexibilität der Storage-Verwaltung dramatisch und ist notwendig, um Features wie Live-Migration virtueller Maschinen sinnvoll zu nutzen.

## Konzept und Terminologie

Das Storage-Konzept von Proxmox VE orientiert sich mit Pools und Volumes an Konzepten anderer Virtualisierungsplattformen wie etwa vSphere. Eine

wichtige Rolle im Storage-Konzept spielen PVE Types und Content Types.

## PVE Type

Der PVE Type ist in Proxmox VE fest definiert und stellt eine Kombination aus Speichertechnik und Anbindungsart (Tabelle 1) dar. Jedem Storage Pool muss einer dieser Typen zugewiesen werden. Vom PVE Type hängt es ab, welche Features der Storage unterstützt, wie komplex die Konfiguration wird und wie hoch die Anforderungen an die Hardware sind. In Produktivsystemen ist es in der Regel sinnvoll, je nach Verwendungszweck Pools

mit unterschiedlichen PVE Types anzulegen. So könnten etwa Backups und ISO-Images per NFS auf ein externes NAS ausgelagert oder Windows-Shares per CIFS angebunden werden. Für Images virtueller Maschinen sollte hingegen möglichst schneller Storage verwendet werden, der Features wie Live-Migration, Snapshots und Thin Provisioning unterstützt und nach Möglichkeit fehlertolerant ist.

## Content Types

Content Types legen den Verwendungszweck eines Storage Pools fest. Pro Pool können bei der Konfiguration mehrere Content Types angegeben

werden, aber nicht jeder PVE Type unterstützt alle Content Types. Eine Liste aller Content Types in Proxmox VE zeigt Tabelle 2.

Tabelle 1: PVE Types

Description	Plugin type	Level	Shared	Snapshots	Stable
ZFS (local)	zfspool	both <sup>1</sup>	no	yes	yes
Directory	dir	file	no	no <sup>2</sup>	yes
BTRFS	btrfs	file	no	yes	technology preview
NFS	nfs	file	yes	no <sup>2</sup>	yes
CIFS	cifs	file	yes	no <sup>2</sup>	yes
Proxmox Backup	pbs	both	yes	n/a	yes
GlusterFS	glusterfs	file	yes	no <sup>2</sup>	yes
CephFS	cephfs	file	yes	yes	yes
LVM	lvm	block	no <sup>3</sup>	no	yes
LVM-thin	lvmthin	block	no	yes	yes
iSCSI/kernel	iscsi	block	yes	no	yes
iSCSI/libiscsi	iscsidirect	block	yes	no	yes
Ceph/RBD	rbd	block	yes	yes	yes
ZFS over iSCSI	zfs	block	yes	yes	yes

<sup>1</sup>: Disk images for VMs are stored in ZFS volume (zvol) datasets, which provide block device functionality.

<sup>2</sup>: On file based storages, snapshots are possible with the *qcow2* format.

<sup>3</sup>: It is possible to use LVM on top of an iSCSI or FC-based storage. That way you get a shared LVM storage

Tabelle 2: Content Types

Content Type	Beschreibung
images	Images virtueller Maschinen in den Formaten raw, qcow2 oder vmd
rootdir	für Containerdaten im Dateisystem des Containers
zvtmpl	Container Templates
backup	Backup Files im vmdump-Format
iso	ISO Images
snippets	Snippet-Dateien, Hook Scripts für Gastssysteme

# Image-Formate

Proxmox VE unterstützt drei Formate für Images von virtuellen Maschinen: Das Raw Disk Images (raw), QEMU Image Format (qcow2) und das VMware Image Format (vmdk) für den Import und Export von vSphere-Images. Bei Block-Storage ist nur das Raw-Format erlaubt, auf File-basiertem Storage sind beide Formate möglich. Um Snapshots, Thin Provisioning und Cloning bei Raw Images zu nutzen, muss das Storage Device, also der PVE Type, diese Features unterstützen. Dabei sind immer dort, wo Snapshots möglich sind, auch Thin Provisioning und Cloning möglich (siehe Tabelle 2). Bei File-Storage,

das von Haus aus dazu nicht in der Lage ist, ist es dennoch nutzbar, wenn das QEMU Image Format verwendet wird.

Thin Provisioning sorgt für effiziente Speichernutzung, da nur tatsächlich vorhandene Daten der VMs Platz belegen. Man kann also einen Storage-Bereich anlegen, den das Gastsystem zum Beispiel als 128 GB-Festplatte sieht. Wenn aber dort nur 10 GB belegt sind, wird auch nur dieser Platz auf dem physischen Storage Device in Anspruch genommen.

## Storage Replication

In Proxmox-Clustern, in denen die Images von VMs nicht auf Shared Storage, sondern lokal gespeichert werden, kann Storage Replication für eine höhere Datensicherheit sorgen. Damit werden in regelmäßigen Abständen Snapshots der VM erzeugt und auf den lokalen Storage eines anderen

Nodes transportiert. Da Snapshots inkrementell sind, werden nach der ersten Replikation im Normalfall nur noch wenige Daten übertragen. Das Feature setzt lokale ZFS-Dateisysteme auf Quell- und Zielhost voraus.

## 7. Hochverfügbarkeit

Hohe Ausfallsicherheit gehört zu den wichtigsten Zielen beim Betrieb von Servern. Cluster mit virtuellen Maschinen, wie sie durch Proxmox VE realisiert sind, bieten prinzipiell viele Möglichkeiten an, Hochverfügbarkeit (High Availability bzw. HA) darzustellen. Meist ist die Konfiguration sehr komplex und kann bei unerfahrenen Administratoren eher zu Datenverlust als zu höherer Verfügbarkeit führen. Proxmox VE besitzt mit dem HA-Manager ein einfach zu handhabendes Interface, um schnell und relativ risikoarm hochverfügbare Umgebungen zu erstellen. Es kann sowohl über das Web-Interface als auch an der Kommandozeile bedient werden. Der HA-Manager sorgt automatisch dafür, dass

beim Ausfall eines Nodes ein anderer Node den Betrieb übernimmt und automatisch die VMs des ausgefallenen Nodes dort startet. Voraussetzung dafür sind mindestens drei Nodes im Cluster und Shared Storage. Eine Ausnahme stellt ein Proxmox VE & ZFS 2 Node-Cluster mit asynchroner Replikation und externem Quorum dar.

Um sich mit den erforderlichen Schritten vertraut zu machen, ohne zu Testzwecken drei Proxmox-Nodes zu installieren, stellt Proxmox sogar einen HA-Simulator als Software-Paket zur Verfügung (Abbildung 3). Außerdem sind sämtliche Konzepte, Schritte und Konfigurationsmöglichkeiten im Administration Guide ausführlich dokumentiert.

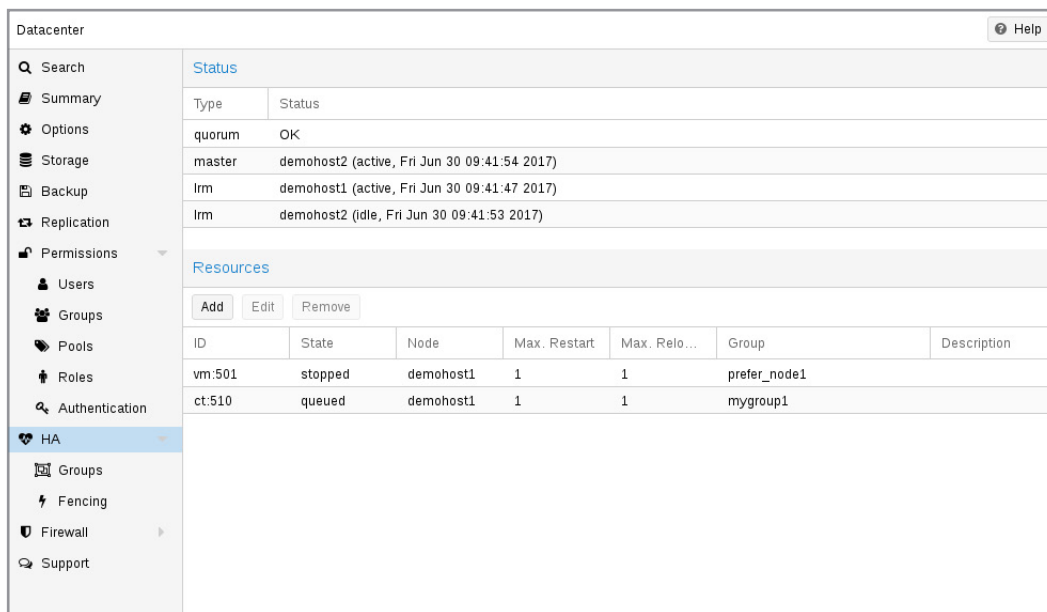


Abbildung 3: HA-Simulator zum Testen von HA-Funktionen ohne eigene Hardware

## 8. Proxmox VE administrieren

Für die Administration von Proxmox VE steht eine webbasierte Oberfläche und eine Reihe Proxmox-spezifischer Kommandozeilen-Werkzeuge zur Verfügung. Dazu kommen sämtliche Linux-Befehle, die bei jeder Debian-Installation vorhanden sind. Zur Zugriffsteuerung auf den Hypervisor bietet Proxmox VE ein typisches rollenbasiertes Zugriffsmodell (RBAC). Die Nutzer-Authentisierung kann mittels einfacher Anmeldung per Passwort, SSH-Keys oder

zwei Methoden der Zwei-Faktor-Authentisierung (TOTP und YubiKey OTP) erfolgen. Rollen sind Sammlungen von Berechtigungen. Berechtigungen gelten für bestimmte Objekte, wie Hosts, VMs, Storage Pools oder einzelne Storages. Neue Rollen lassen sich schnell definieren, so dass auch in größeren Enterprise-Umgebungen ein komplexes, feingranulares Berechtigungsmanagement für Administratoren möglich ist.

### Grafisches User-Interface

Proxmox VE speichert die Konfigurationsdateien aller Nodes in einem über den gesamten Cluster verteilten Dateisystem. Jeder Node stellt ein Web-GUI zur Konfiguration zur Verfügung, somit ist die Cluster-Administration von jedem Node aus möglich. Eine zentrale Verwaltungsinstanz wie vCenter bei VMware ist nicht notwendig. Die Web-GUI stellt den aktuellen Zustand der Nodes und VMs grafisch dar (Abbildung 4) und dient zum schnellen Erstellen von VMs, Containern,

virtuellen Netzwerken, Storage-Bereichen usw. Voraussetzung ist, dass ein Anwender gut mit den Proxmox VE Konzepten vertraut ist. Deshalb wird gerade für Einsteiger empfohlen, sich auch mit den Kommandozeilen-Werkzeugen vertraut zu machen.

Die Web-GUI kommt mit eingebauter grafikfähiger Konsole für den Zugriff auf Host, VMs und Container. Die Installation und Verwaltung von Windows-Gast-Systemen stellt somit kein Problem dar. Eine SPICE-



Konsole kann die Arbeit auf grafischen Oberflächen weiter verbessern, erfordert aber entsprechende Treiber für Gast-Systeme, die unter Proxmox VE einfach installiert werden können.

Obwohl die Proxmox-Macher mit responsivem Design der Web-GUI werben, ist sie auf kleinen Handy-Displays nur eingeschränkt nutzbar. Für kleinere Administrationsarbeiten im Notfall, wie

das Starten oder Stoppen von virtuellen Maschinen, reicht sie aber auch dort aus. Zusätzlich existiert eine mobile Variante der GUI mit eingeschränkten Funktionen und eine Android App namens Aprox von einem Drittanbieter (über den Google Play Store verfügbar).

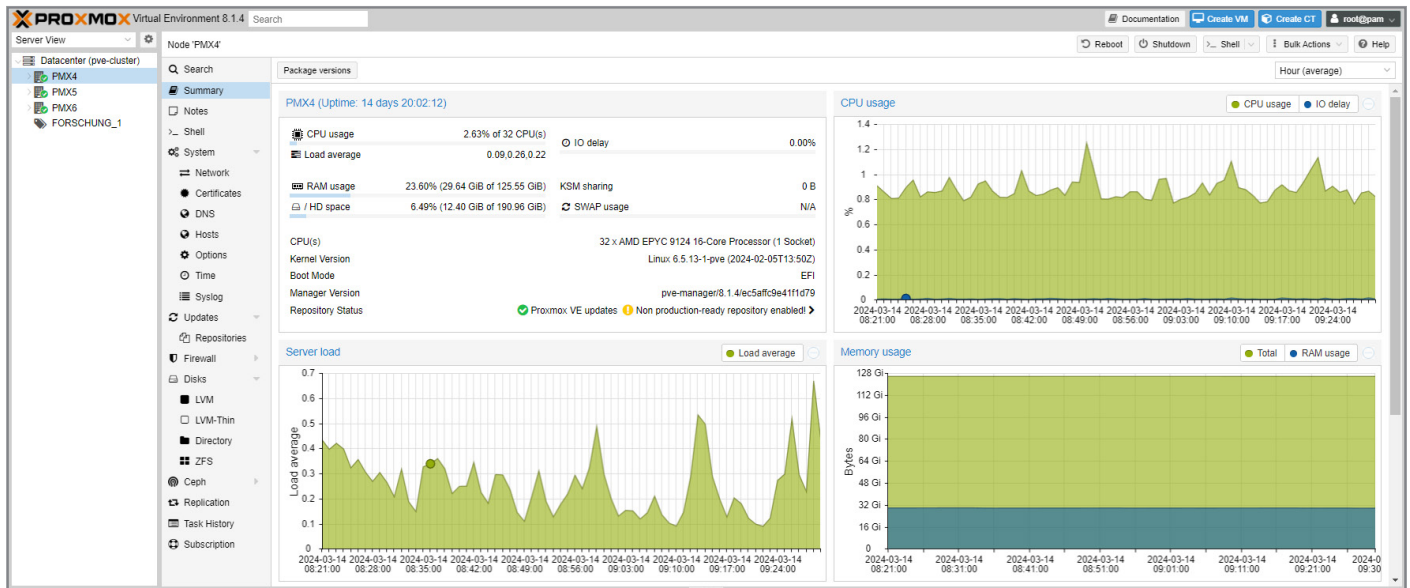


Abbildung 4: Die Web-GUI von Proxmox VE

## Kommandozeilen-Tools

Proxmox-spezifische Kommandozeilen-Werkzeuge (CLI Tools) starten meist mit pve. Neben den in Tabelle 3 genannten existieren noch weitere mit selbsterklärendem Namen, etwa zur Firewall-Konfiguration oder zum Erstellen von Reports. In der Regel geben die PVE-Kommandos beim Aufruf mit pvecommand help in Kurzform Auskunft über ihre Verwendung. Mehr Informationen liefern die

Manpages. Auch der Proxmox VE Administration Guide bringt zahlreiche Beispiele für den Einsatz dieser Kommandos. Für das Tuning von Netzwerk und Storage werden darüber hinaus oft Low Level Tools auf Betriebssystem-Ebene eingesetzt oder Konfigurationsdateien direkt editiert.

Tabelle 3: Wichtige Kommandozeilen-Werkzeuge

Kommando	Beschreibung
pveam	Template-Verwaltung und -Download
pveceph	Verwaltung von Ceph-Nodes und -Pools
pvecm	Cluster-Manager: Anlegen und Konfiguration von Clustern
pvenode	Node-Manager: Node-Konfiguration, Start, Stop, Migration von VMs, Zertifikatsverwaltung
pvesm	Storage Manager: Umfassendes Werkzeug zur Storage-Verwaltung
pvesr	Replikation von virtuellen Maschinen
pveum	Nutzer- und Rechteverwaltung
pvesh	Direkter Zugriff auf API von der Kommandozeile
qm	Umfangreiches Werkzeug zur Administration virtueller Maschinen
pct	Umfangreiches Werkzeug zur Administration von Linux Containern (LXC)

## API

GUI und CLI-Tools greifen über die gleiche API auf die Konfiguration von Nodes und virtuellen Maschinen zu. Mit den CLI-Tools gemachte Änderungen sind also immer auch sofort in der GUI sichtbar und umgekehrt. Darüber hinaus eröffnet die API eine dritte Möglichkeit, Proxmox VE zu administrieren. Durch sie ist es möglich, Proxmox VE in bestehende Frameworks zur Cloud-Orchestrierung einzubinden

oder Tools zur Automatisierung wiederkehrender Aufgaben zu schreiben. Die API folgt dem REST-Standard, nutzt JSON als Übertragungsformat für Konfigurationsdaten und Token-basierte Authentisierung. Die Proxmox Server Solutions GmbH pflegt einen API-Client in der Programmiersprache Perl, für Java, Python, NodeJS, C#, Ruby und PHP existieren Community-Projekte.

## 9. Tipps zur Hardware

Im Gegensatz zu anderen Virtualisierungslösungen ist Proxmox VE auch schon auf kleineren Systemen einsetzbar. Insbesondere gilt das, wenn Gastssysteme nur in leichtgewichtigen Containern betrieben werden. Welche Ressourcen am Ende tatsächlich benötigt werden, ist abhängig von den Ansprüchen der Gast-Systeme, aber auch vom eingesetzten Storage. Vor allem verteilte Systeme

wie ZFS oder Ceph verlangen CPU, RAM und Netzwerk viel ab.

Im Allgemeinen ist es ratsam, für alle Nodes im Cluster identische Hardware einzusetzen. Das gilt besonders, wenn ein hyperkonvergentes System mit Ceph oder ein ausfallsicherer Cluster mit Proxmox VE HA konfiguriert werden soll.

### CPU

Prinzipiell läuft Proxmox VE auf jeder CPU mit Intel VT oder AMD-V-Technik, Hardware-Virtualisierung profitiert aber extrem von CPUs mit vielen Kernen. Bei höheren Lasten empfiehlt sich auch ein Dual-Socket-System. Ceph-Storage stellt hohe Ansprüche an die CPU. Es empfiehlt sich, mindestens einen, besser zwei Threads pro SSD oder HDD zu reservieren, die von Ceph genutzt

wird, zusätzlich zu den Cores, die für die virtuellen Maschinen und das Host-System selbst benötigt werden. Kommt NVMe-Speicher zum Einsatz, muss auch sichergestellt sein, dass genügend PCIe-Lanes zum Verbinden des Storage vorhanden sind. ZFS auf dem Hostsystem verlangt ebenfalls nach hoher Rechenpower. Nach Möglichkeit sollten hier CPUs mit hoher Taktrate zum Einsatz kommen.

### Hauptspeicher

Das Proxmox VE Hostsystem selbst ist nicht speicherhungrig. Zwei oder vier GB reichen aus. Diese werden einfach zum geplanten RAM-Bedarf der virtuellen Maschinen oder Container addiert. Bei Storage-Systemen wie ZFS oder Ceph gilt: Je mehr RAMdestobesser. Bei ZFS liegt man mit der Faustregel von einem GB RAM pro TB Brutto-Speicherplatz plus

4 GB für das Speichermanagement auf der sicheren Seite. Für Ceph empfiehlt die Dokumentation 3 GB bis 5 GB RAM pro SSD, hinzu kommen je nach Cluster-Größe 2 GB bis 10 GB pro Node für Cluster-Management und Metadaten. [https://pve.proxmox.com/pve-docs/pve-admin-guide.html#\\_recommendations\\_for\\_a\\_healthy\\_ceph\\_cluster](https://pve.proxmox.com/pve-docs/pve-admin-guide.html#_recommendations_for_a_healthy_ceph_cluster)

### Netzwerk

In Produktivsystemen sollten mindestens zwei physische Netzwerk-Interfaces pro Node vorhanden sein, um den Traffic, der beim Cluster-Management, bei Replikation und Migration von VMs entsteht, von der "Nutzlast", also dem Datenverkehr der virtuellen Maschinen und Container selbst, zu trennen. Mehr Netzwerkkarten erhöhen die Flexibilität im Betrieb. Beispielsweise lassen sich durch Bonding eventuell

aufgetretene Engpässe vermeiden. ZFS und insbesondere Ceph tragen wesentlich zur Netzwerk-Bandbreite bei, hier sollte das Cluster-Interconnect mit mindestens mit 10 Gbit-Interfaces, bei Ceph besser 25 Gbit, ausgestattet sein. Die konkreten Anforderungen hängen jedoch immer sehr stark vom individuellen Verwendungszweck ab.

## RAID-Controller

Wer ZFS oder Ceph für das Storage einsetzt, benötigt keine RAID-Controller. Deren Funktionalität (und mehr) wird vollständig in der Software abgebildet. Vorhandene RAID-Controller müssen im Passthrough-Modus ohne eigenes RAID-Layer betrieben werden (bei Broadcom als IT-Modus bezeichnet). Sonst sind Host Bus Adapter die

bessere Wahl.

Das Proxmox-Wiki bietet umfangreiche Informationen zu RAID-Hardware und deren Einrichtung.

[https://pve.proxmox.com/wiki/Raid\\_controller](https://pve.proxmox.com/wiki/Raid_controller)

## Proxmox VE Ceph HCI

Beim Aufbau einer Hyperconverged Infrastructure (HCI) mit Proxmox VE und Ceph empfiehlt sich der Einsatz von speziell dafür entwickelten Nodes und darauf abgestimmter Netzwerk-Komponenten, inklusive Herstellersupport. Thomas-Krenn bietet mit Proxmox VE Ceph HCI (all NVMe) eine kompakte und dennoch leistungsfähige und erweiterbare Lösung an. Trotz des flachen Designs mit nur einer Höheneinheit bieten die Server Platz für redundante Netzteile, jeweils zehn NVMe-SSDs, zwei SATA-SSDs für das Betriebssystem sowie bis zu acht Netzwerk-Ports. Als Prozessoren kommen EPYC-CPU's von AMD auf Single-Socket-Mainboards zum Einsatz.

Dank der hohen Anzahl von Kernen und PCIe-

Lanes bieten sie die Voraussetzungen, um Ceph-basierte HCI-Cluster bereits mit drei Nodes produktiv einsetzen zu können. Bei Bedarf lassen sich die Cluster auf bis zu 32 Nodes erweitern. Je nach Anforderung können Anwender zwischen Netzwerk-Komponenten von 10 GBit/s bis 100 GBit/s und Switches von Netgear oder Mellanox wählen. Erstinstallation und Inbetriebnahme wird bei Bedarf von Thomas-Krenn übernommen. Optional empfiehlt sich der Erwerb von Proxmox VE-Subskriptionen für Support durch die Proxmox Server Solutions GmbH.

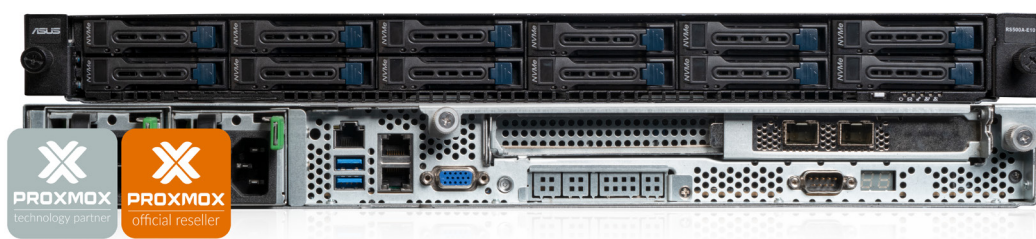


Abbildung 5: Neben zahlreichen Proxmox-optimierten Systemen bietet Thomas-Krenn mit Proxmox Ceph HCI auch eine Lösung speziell für Software Defined Storage und hyperkonvergente Infrastrukturen an.

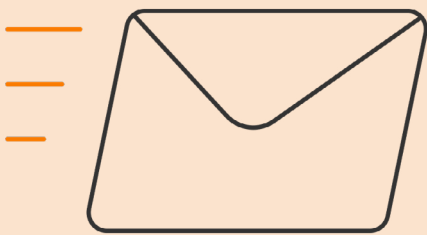
## 10. Fazit

Mit seiner Kombination von Hardware- und Container-Virtualisierung hat Proxmox VE unter allen verfügbaren Virtualisierungsplattformen ein echtes Alleinstellungsmerkmal. In sich konsistente Konzepte und Werkzeuge vereinheitlichen die Verwaltung von virtuellen Maschinen, Containern, Storage und Netzwerk. Insbesondere häufig auftretende Anwendungsfälle wie die Bereitstellung hochverfügbarer Dienste sind mit Proxmox VE schnell umgesetzt. Die optimale Storage-Anbindung setzt wegen der Vielzahl der unterstützten Interfaces sorgfältige Planung voraus.

Durch die Integration von ZFS und Ceph sind auch anspruchsvolle fehlertolerante Storage-Szenarien und Hyperkonvergenz möglich. Für Performance-Tuning und komplexere Konfigurationen sind Administratoren mit guten Linux-Kenntnissen von Vorteil. Die umfangreiche Dokumentation, eine

große Community und nicht zuletzt der Support durch die Proxmox Server Solutions GmbH bieten bei Konzeption und im praktischen Betrieb gute Unterstützung.

Die Auswahl geeigneter Hardware für Proxmox VE hängt stark von den Ansprüchen der Anwendungen und vom verwendeten Storage ab. Jedoch führt schon der Einsatz auf kleineren Systemen oft zu reduziertem Administrationsaufwand und damit zu Kostensenkungen in der IT. Im Vergleich zu proprietären Plattformen ist dabei das Risiko eines Vendor Lock-Ins mit hohen Folgekosten beim Systemwechsel gering. Auch sind Kostenfallen beim Ausbau der Plattform, etwa durch separate Lizenzierung bestimmter Features, bei dieser Open Source Software ausgeschlossen.



Sie möchten keine **Neuheiten, Trends oder wichtige Tipps** verpassen? Dann abonnieren Sie unseren **Newsletter** und wir halten Sie auf dem Laufenden!

**Newsletter abonnieren**



**Sie haben Fragen/Anregungen/Feedback zu unserem E-Book?**

**Wir freuen uns auf den Austausch mit Ihnen - [Schreiben Sie uns gerne an!](#)**

Ihnen gefällt, was Sie lesen?

Mehr E-Books gibt's hier:

[www.thomas-krenn.com/ueberblick\\_ebooks](http://www.thomas-krenn.com/ueberblick_ebooks)

Thomas-Krenn.AG  
Speltenbach-Steinäcker 1  
D-94078 Freyung  
[thomas-krenn.com](http://thomas-krenn.com)

**THOMAS  
KRENN<sup>®</sup>**  
IT's people business