

Marco Gabriel



Proxmox VE

Das umfassende Handbuch

- ▶ Virtualisierte Infrastrukturen aufbauen, migrieren und verwalten
- ▶ Backup & Restore, Hochverfügbarkeit und Lastenmanagement
- ▶ Alles für zuverlässige Cluster: hyperkonvergenter Storage mit Ceph, Live-Migrationen, Software-Defined Networking



Rheinwerk
Computing

Kapitel 4

Installation und Updates

Ein Grund für den Erfolg von Proxmox VE ist sicherlich der einfache Einstieg, und dazu trägt die Installation bei. In diesem Kapitel schauen wir uns die Installation und die Durchführung von Updates an.

4.1 Installation

Möglicherweise haben Sie selbst schon mal die Installation durchgeführt und sich wenige Minuten später darüber gefreut, dass es wirklich so einfach ist. Der Installer kann jedoch mehr, und außerdem kann Proxmox VE nicht nur mit dem offiziellen ISO-Image manuell installiert werden.

Sie können für die Installation zwischen verschiedenen Vorgehensweisen wählen:

■ Installation mit dem ISO-Image von Proxmox VE

Sie verwenden das offizielle ISO-Image von Proxmox VE und installieren einen Proxmox-VE-Server in wenigen Schritten. Das ISO-Image enthält bereits ein vollwertiges Debian Linux. Verwenden Sie nach Möglichkeit diese Variante.

Nach dem Start des ISO-Images können Sie sich entscheiden, ob Sie den Installer im grafischen oder im textbasierten Modus (*Text User Interface, TUI*) verwenden möchten. Das TUI kam mit Version 8.0 neu hinzu und ist vor allem dann interessant, wenn Ihr Server eine nicht unterstützte Grafikkarte besitzt, wenn Ihnen der Textmodus lieber ist oder wenn Sie sonstige Schwierigkeiten mit dem grafischen Installer haben. Die Funktionen und Schritte beider Varianten sind gleich.

Seit Version 8 lässt sich die Installation mit dem offiziellen ISO-Image sogar einfach automatisieren. Das lohnt sich vor allem in zwei Szenarien: Wenn Sie regelmäßig neue Server installieren, beispielsweise in einem Datacenter oder als Hardwarelieferant, oder wenn Sie absolut sicher gehen möchten, auf allen Servern die exakt gleiche Konfiguration angewendet zu haben.

■ Installation über Debian mit Proxmox-VE-Repositories

Immer dann, wenn Sie bei der Installation jedoch etwas benötigen, das der Proxmox-VE-ISO-Installer nicht anbietet, können Sie auch alternativ ein vollständiges Debian installieren und anschließend die Proxmox-VE-Repositories einbinden und die Proxmox-VE-Pakete nachinstallieren.

Beispiele hierfür sind eine vom Standard abweichende Partitionierung oder die Einbindung von Hardware, die vom Proxmox-VE-Installer nicht erkannt oder unterstützt wird.

Achten Sie auf den Support

Nicht jedes Feature einer Debian-gestützten Installation ist vom Proxmox-Support abgedeckt. Beispielsweise ist ein *mdraid* (also ein Linux-Software-RAID) explizit davon ausgenommen. Wenn Sie ein Software-RAID verwenden möchten, machen Sie es einfach wie die coolen Kids und verwenden Sie das von Proxmox supportete ZFS.

4.1.1 Systemvoraussetzungen

Eine allgemeine Empfehlung für Produktivumgebungen ist nicht einfach zu geben. Proxmox VE wird von ganz kleinen Umgebungen mit nur wenigen, kleinen VMs bis hin zu größeren Cluster-Systemen mit Dutzenden Servern und Tausenden VMs verwendet. Vor allem die Ressourcen für die VMs sind an dieser Stelle natürlich nicht anzugeben; ich kann nur entsprechende Herleitungen bieten, damit Sie Ihre Hardware ordentlich auswählen und dimensionieren können.

Mindestvoraussetzungen zur Evaluierung

Falls Sie sich den Hypervisor nur mal ansehen möchten, geht das auch schon auf sehr schmal ausgestatteten Maschinen. Sie brauchen in diesem Fall:

- eine 64-Bit-CPU (Intel x86-64 oder AMD64)
- eine Intel-VT- oder AMD-V-fähige CPU und ein ebensolches Mainboard für eine vollständige KVM-Unterstützung
- mindestens 1 GB RAM für den Hypervisor zuzüglich RAM für virtuelle Maschinen
- mindestens 10 GB Disk Speicher für Proxmox VE und weiteren Speicher für virtuelle Maschinen
- eine 1-Gbit/s-Netzwerkkarte

Empfohlene Systemvoraussetzungen für den Produktivbetrieb

Für einen Produktivbetrieb darf es dann schon etwas mehr Ausstattung sein:

- eine 64-Bit-CPU (Intel x86-64 oder AMD64).
- eine Intel-VT- oder AMD-V-fähige CPU und ein ebensolches Mainboard für eine vollständige KVM-Unterstützung.
- mindestens 2 GB RAM für den Hypervisor zuzüglich RAM für virtuelle Maschinen. Beim Einsatz von Ceph- oder ZFS-Storage wird weiterer RAM benötigt.
- schnelle und redundante Storage-Systeme sind ideal. Die besten Ergebnisse werden Sie mit SSDs oder NVMeS erzielen.

- Storage für das Betriebssystem: 100 GB redundanter Disk-Speicher in einem Hardware-RAID1 oder einem ZFS-Mirror aus NVMe-Devices, SSDs oder drehenden Disks. Verwenden Sie keine USB-Sticks und SD-Karten als Installationsmedium. Da Proxmox auch auf der System-Disk viele Schreibzugriffe durchführt, werden auch gute USB-Sticks und SD-Karten relativ schnell kaputtgehen.
- Storage für virtuelle Maschinen:
 - Für lokalen Speicher sollte entweder ein Hardware-RAID-Controller mit BBU/FBU oder ein HBA verwendet werden.
 - Shared Storage mit Ceph, iSCSI, FibreChannel und NFS ist möglich und sollte bei mehreren Proxmox-VE-Servern verwendet werden.
- Mehrere 1-Gbit/s- oder schnellere Netzwerkkarten, abhängig vom weiteren Faktoren wie Storage, der Anzahl an VMs und benötigter Performance
- Für PCI(e)-Passthrough wird VT-d oder AMD-d benötigt.

4.1.2 Installation über den ISO-Installer von Proxmox VE

Beginnen Sie mit dem Download des offiziellen ISO-Images von der Proxmox-Homepage. Da es sich um ein Hybrid-ISO-Image handelt, können Sie es entweder direkt auf eine CD oder DVD oder auf einen USB-Stick schreiben und anschließend von diesem Medium booten.

Gehen wir die Installation Schritt für Schritt durch:

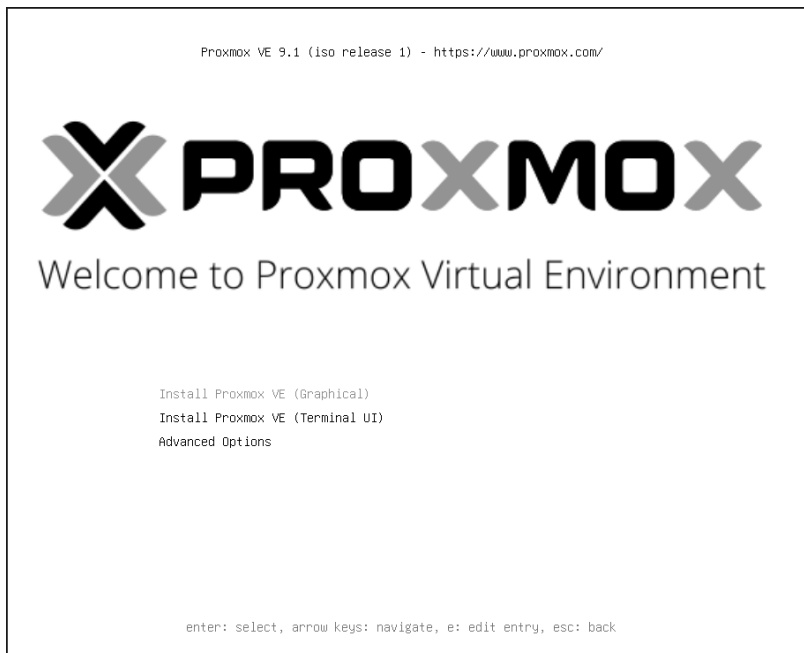


Abbildung 4.1: Nach dem Start erscheint das Bootmenü.

Zunächst bootet das ISO-Image in ein Menü, in dem Sie wie in Abbildung 4.1 zwischen dem grafischen Installer (Standard), der Installation im Textmodus (TUI) oder erweiterten Funktionen (**Advanced Options**) wählen können. Meistens möchten Sie hier den Standard beibehalten und dem grafischen Installer folgen.



Abbildung 4.2: Wählt man »Advanced Options«, erscheinen weitere Möglichkeiten.

Im optionalen Menü **Advanced Options** finden Sie folgende Möglichkeiten (siehe Abbildung 4.2):

- **Install Proxmox VE (Graphical, Debug Mode):** Verwenden Sie den grafischen Installer im Debug-Modus, wenn Sie während der normalen Installation auf Probleme stoßen und diese näher untersuchen möchten. Der Debug-Modus geht den gleichen Weg wie der normale Modus, hält aber an verschiedenen Punkten während der Installation an, und Sie können zusätzliche, manuelle Untersuchungen oder Änderungen durchführen.
- **Install Proxmox VE (Terminal UI, Debug Mode):** Die Installation erfolgt wie beim Debug-Modus des grafischen Installers, aber eben im Textmodus.
- **Install Proxmox VE (Serial Console Debug Mode):** Wenn Sie keinerlei Ausgabe auf dem Bildschirm erhalten oder vielleicht sogar keine Grafikkarte oder keinen Bildschirm haben, erlaubt dieser Modus die Installation und den Zugriff über eine serielle Konsole.
- **Install Proxmox VE (Automated):** Dieser Punkt erlaubt den Start des ISO-Images für die automatische Installation anhand einer vorher festgelegten Konfiguration. Sie

müssen danach keine Angaben mehr machen, da die Konfiguration bereits vorab festgelegt wird.

- **Rescue Boot:** Erlaubt es, einen nicht mehr startenden Server mit einem temporären System zu booten, um anschließend eine Fehleranalyse oder Reparatur vorzunehmen.
- **Test memory (memtest86+):** Der übliche Speichertest, mit dem Sie bei Bedarf den RAM auf Fehler testen können.

Beim Bootvorgang sehen Sie diverse Meldungen sowie einen DHCP-Request des Servers. Nur der Installer verwendet DHCP, der Server selbst erhält eine fest eingetragene IP-Adresse während der Installation. Obwohl der Server nun möglicherweise eine funktionierende Internetverbindung genießt, wird die Installation nur mit den Paketen aus dem ISO-Image durchgeführt. Während der Installation werden keine Updates oder sonstigen Pakete aus dem Internet heruntergeladen.

Reproduzierbare Installationen

Dies ermöglicht Ihnen übrigens auch, jederzeit reproduzierbare Installationen durchzuführen, solange Sie das gleiche ISO-Image einsetzen.

Nach dem Start der Installation müssen Sie zunächst den Lizenzvertrag (EULA) bestätigen (siehe Abbildung 4.3). Mehr zu der Lizenzierung von Proxmox VE erfahren Sie in Kapitel 13, »Subscriptions«.

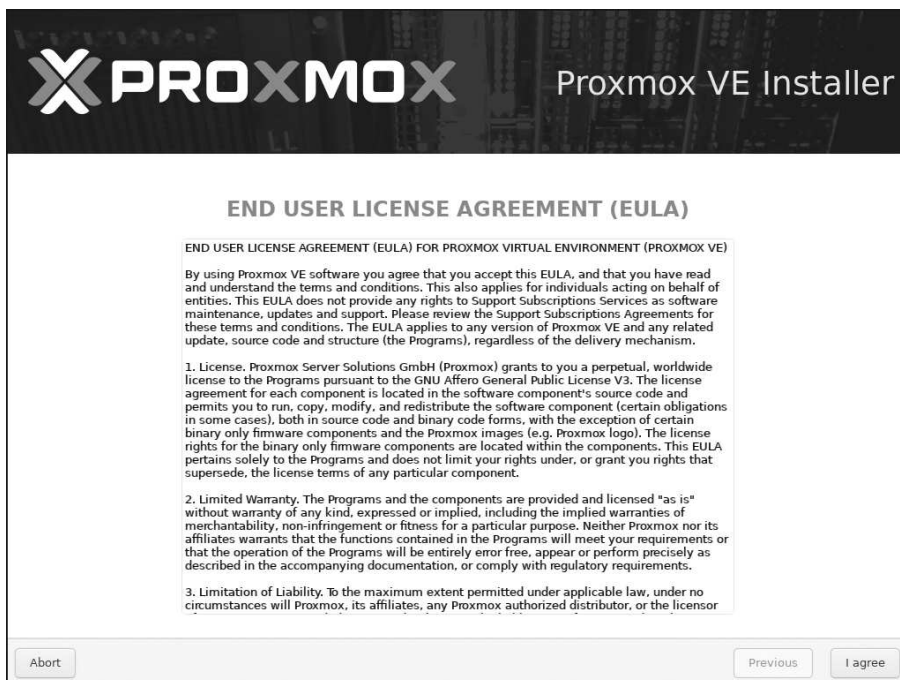


Abbildung 4.3: Seite 1 – Bestätigen des End-User-License-Agreements (EULA)

In Virtualisierungsservern sind häufig mehrere lokale Disks eingebaut. Für die Installation sind jedoch nur die Disks wichtig, auf denen Proxmox VE installiert werden soll. Nehmen wir an, Sie haben einen Server mit zwei 500-GB-Disks für das System und zwei 8-TB-Disks für die VMs und Container. Für diesen Schritt sind nur die beiden 500-GB-Disks relevant. Ihren Datenspeicher aus den beiden 8-TB-Disks erstellen Sie erst später, wenn der Server fertig installiert und betriebsbereit ist.

Auf Seite 2 (siehe Abbildung 4.4) ist es entscheidend, ob Sie einen Hardware-RAID-Controller oder ein ZFS/btrfs-Software-RAID verwenden. Sie können auf dieser Seite nur eine einzelne Disk angeben, auf die Proxmox VE installiert werden soll. Gehen Sie am besten so vor, wie ich es in den folgenden beiden Abschnitten beschreibe.

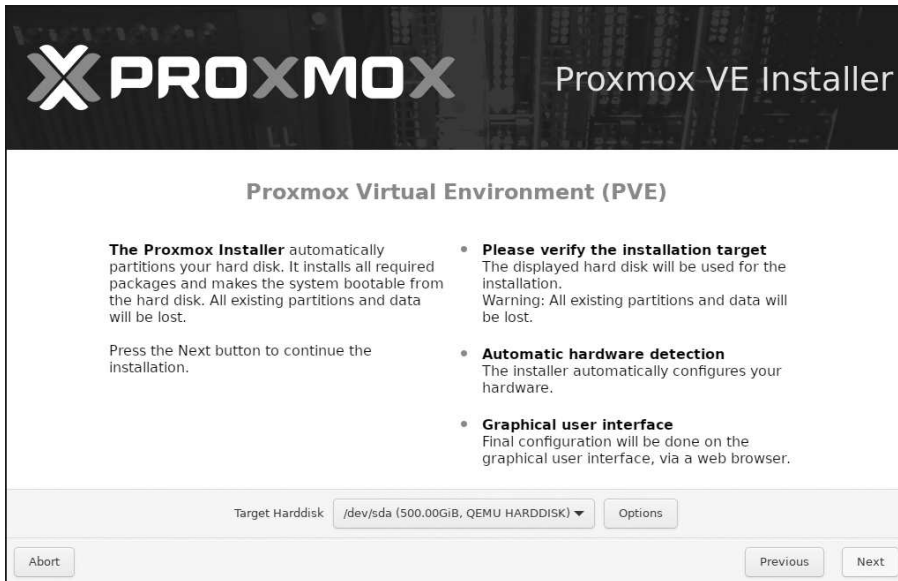


Abbildung 4.4: Seite 2 – Auswahl der Disk(s), auf denen die Installation ausgeführt werden soll

Hardware-RAID

Bei einem Hardware-RAID haben Sie bereits vorher im RAID-Controller Ihre Disks zu einem RAID-Volume konfiguriert. Wählen Sie hier das im RAID-Controller erstellte RAID-Volume aus, das Sie für Ihr Betriebssystem vorgesehen haben.

Wählen Sie hier nur das RAID-Volume oder eine einzelne Disk aus, wird dieses Volume als *LVM Physical Volume (PV)* verwendet. In diesem PV wird eine *Volume Group (VG)* namens *pve* erstellt, und darin werden drei verschiedene *Logical Volumes (LV)* mit den Namen *root*, *data* und *swap* angelegt.

Das Volume *root* wird mit *ext4* formatiert und das Betriebssystem wird auf ihm installiert. Auf *swap* wird ein Auslagerungsspeicher bereitgestellt, und auf *data* werden später die Images Ihrer VMs und Container abgelegt. Das Logical Volume *data* wird vom Installer als *LVM Thin Volume* bereitgestellt.

Im Dialog können Sie zusätzlich folgende Einstellungen vornehmen, die für eine LVM-basierte Installation relevant sind:

- **hdszise** gibt den maximal zu verwendenden Speicher des zugrunde liegenden Volumens an. Haben Sie beispielsweise nur zwei 8-TB-Disks, können Sie diesen Wert auf 100 GB setzen. Dann werden nur diese 100 GB für die Installation verwendet. Die restlichen 7,9 TB stehen Ihnen damit für das Erstellen weiterer Datenvolumen zur Verfügung.
- **swapszise** gibt an, wie groß der Auslagerungsspeicher maximal dimensioniert sein soll. Gibt man hier keinen Wert an, ermittelt der Installer anhand der RAM- und Disk-Größe einen Wert zwischen 512 MB und 8 GB.
- **maxroot** legt fest, wie viel Speicher dem root-Filesystem maximal zur Verfügung stehen soll. Wird kein Wert angegeben, ermittelt Proxmox anhand des verfügbaren Speichers selbstständig einen Wert. Bei den Ressourcen heutiger Hypervisoren liegt die Größe des root-Volumens zwischen 25 % und 33 % der gesamten Disk-Größe.
- **minfree** gibt an, wie viel Speicher der Installer in der LVM-Volume-Group frei lassen soll. In älteren Versionen von Proxmox VE mit LVM *thick Volumes* war dies notwendig, um Storage-Snapshots zu ermöglichen.
- **maxvz** legt die maximale Größe für das Volume **data** fest.

In Abbildung 4.5 sehen Sie am Beispiel eines Servers mit zwei 500-GB-System-Disks und zwei 8-TB-Daten-Disks das Zusammenwirken der Komponenten.

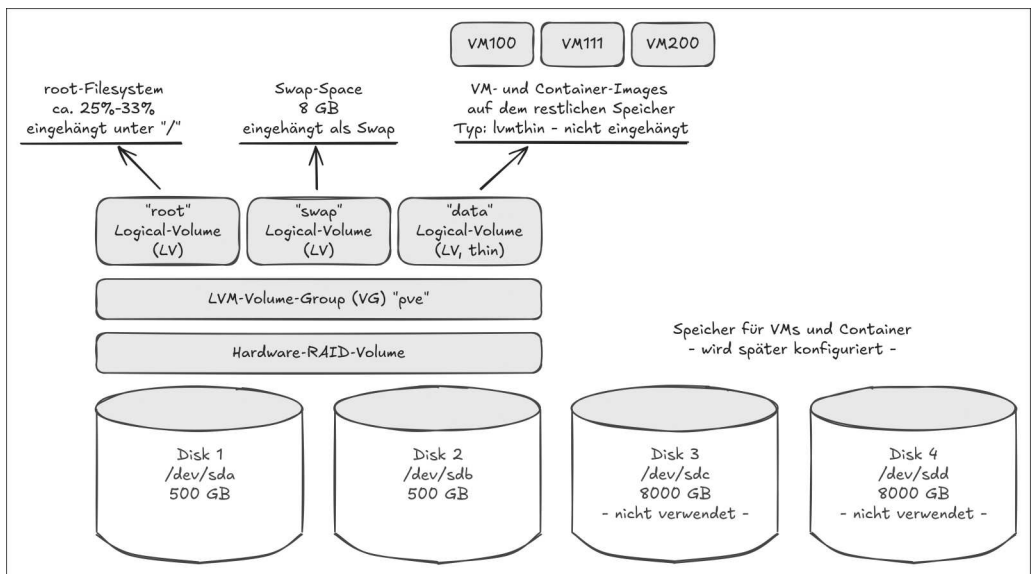


Abbildung 4.5: Datenspeicher-Layout bei LVM-basierter Installation

Unten sehen Sie die Disks, auf denen ein RAID-Volume durch den Hardware-RAID-Controller bereitgestellt wird. Dieses wird als Physical Volume für LVM eingerichtet und eine LVM-Volume-Group wird darin angelegt. In der Volume Group finden Sie dann die Logical Volumes **root**, **swap** und **data**. **data** ist als *LVM thin Volume* angelegt und nimmt die VM- und Container-Images auf.

Software-RAID mit ZFS/btrfs

Wenn Sie einen HBA statt eines RAID-Controllers im Server verwenden, sollten Sie ein Software-RAID auf Basis von ZFS oder btrfs nutzen. Über einen Klick auf **Options** (siehe Abbildung 4.4) gelangen Sie in einen Pop-up-Dialog mit weiteren Möglichkeiten, den Sie in Abbildung 4.6 sehen.



Abbildung 4.6: Optional können Sie die Filesystem- und Software-RAID-Konfiguration bestimmen.

Zu Beginn ist im Dialog **ext4** als **Filesystem** ausgewählt. Sie können hier andere Dateisysteme wie **xfs**, **zfs** und **btrfs** für das root-Filesystem wählen:

- ext4
- xfs
- zfs (RAID0)
- zfs (RAID1)
- zfs (RAID10)
- zfs (RAIDZ-1)
- zfs (RAIDZ-2)
- zfs (RAIDZ-3)
- btrfs (RAID0)
- btrfs (RAID1)
- btrfs (RAID10)

Sobald Sie sich für ein Filesystem mit angegebenem RAID-Level entscheiden (siehe Abbildung 4.7), verändert sich das Dialogfenster und Sie erhalten passende Auswahlmög-

lichkeiten für die Disks, auf denen der Hypervisor installiert werden soll (siehe Abbildung 4.8).



Abbildung 4.7: Filesystem und Software-RAID-Konfiguration bestimmen, Auswahlmöglichkeiten

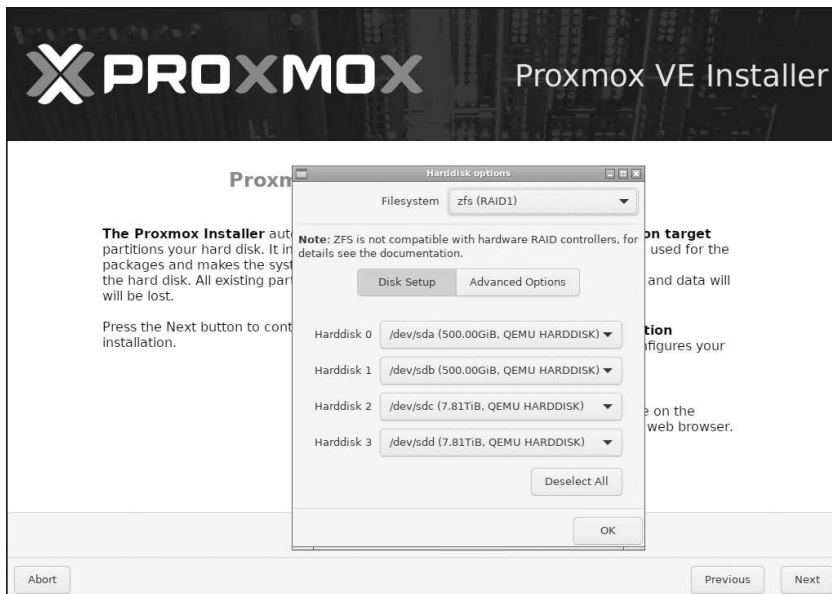


Abbildung 4.8: ZFS – RAID-Level und Disks auswählen

Wählen Sie am besten zuerst alle Disks ab und suchen Sie danach ausschließlich die Disks aus, auf die das Proxmox-VE-System installiert werden soll. Alle ausgewählten Disks

werden bei der Installation gelöscht. Deshalb prüfen Sie bei einer Neuinstallation, ob Sie die richtigen Disks ausgewählt haben.

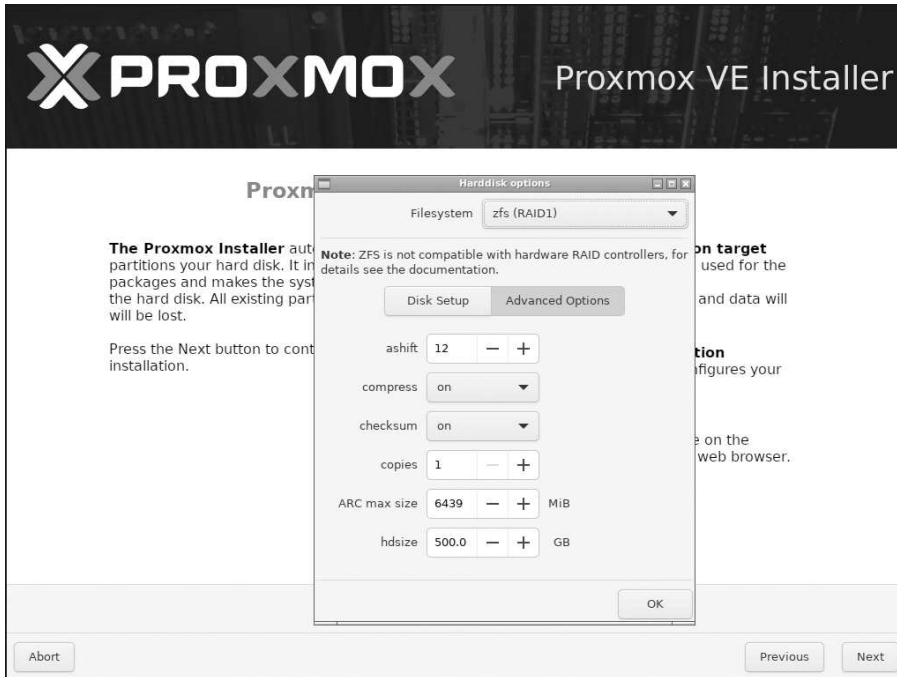


Abbildung 4.9: ZFS – Advanced Options

Die Wahl von ZFS erlaubt es Ihnen, einige Parameter anzupassen (siehe Abbildung 4.9):

- **ashift** gibt das *Alignment* an, also die Ausrichtung von Sektoren auf den Datenträgern. Der Default-Wert von 12 entspricht $2^{12} = 4096$ Byte oder 4k Byte. Die meisten Festplatten, SSDs und NVMe-Devices verwenden heute eine Sektorgröße von 4k Byte, so dass dieser Wert in den allermeisten Fällen passt. Falls Ihr Device eine größere Sektorgröße verwendet, wie 8k oder gar 16k, setzen Sie diesen Wert auf 13 oder 14 hoch.
- **compress** schaltet die integrierte Datenkomprimierung von ZFS ein oder aus. Sie können die Komprimierung auch später jederzeit deaktivieren und wieder aktivieren, aber bedenken Sie, dass ZFS die Komprimierung auch für *Sparse Files* verwendet. Der voreingestellte Komprimierungsalgorithmus ist **lz4**, Sie können ihn jedoch später jederzeit anpassen.
- **checksum** bestimmt, welcher Algorithmus zum Berechnen der Prüfsummen im ZFS-Root-Pool verwendet wird.
- **copies** bestimmt die Anzahl der Kopien, die ZFS schreibt. Es ist kein Ersatz für Redundanz, jedoch kann es bei extrem kritischen Daten die Wahrscheinlichkeit erhöhen, mit

der Daten wiederherstellbar sind. In der Regel werden Sie diese Einstellung beim Standardwert 1 belassen, insbesondere beim Root-Pool.

- **ARC max size** bestimmt die Größe des *Adaptive Read Cache* (ARC) von ZFS und kann seit Version 8 direkt im Installer hinterlegt werden. Sie erhalten hier einen errechneten Vorschlagswert, den Sie übernehmen oder ändern können. Der ARC-Cache steht nicht nur für den root-Pool bereit, sondern auch für alle anderen Pools im System. Wenn Sie weitere Disks erst später in Betrieb nehmen, sollten Sie diesen Wert am besten gleich anpassen. Pro TiB nutzbarem Disk-Speicher in Ihrem Server sollten Sie hier 1 GiB oder 1024 MiB eintragen. Hat Ihr Server alles in allem 50 TiB nutzbaren Speicher, tragen Sie in dieses Feld $50 \times 1024 = 51200$ MiB (50 GiB RAM) ein.
- **hdspace** gibt an, wie viel Disk-Speicher verwendet werden soll, und zeigt per Default die gesamte Größe des Device an. Wenn Sie die Größe an dieser Stelle reduzieren, können Sie den freien Speicher später für die Einrichtung einer Swap-Partition oder für andere Zwecke nutzen.

Eine Swap-Partition wird bei der Verwendung eines ZFS nicht automatisch angelegt. Ihr Server verfügt damit zunächst nicht über einen Auslagerungsspeicher. Das ist auch erst einmal gar nicht tragisch, da die meisten Hypervisoren genügend RAM besitzen und gar nicht swappen müssen.

Vorsicht beim Swap-Space auf ZFS

Wenn Sie dennoch einen Swap-Space einrichten möchten, dann legen Sie dafür später einen separaten Pool an! **Legen Sie den Swap nie mit auf den rpool**, da dies bei hoher Systemauslastung zum Stillstand des Servers führen kann!

Wenn Sie **btrfs** statt ZFS aus der Liste der Filesysteme auswählen, stehen Ihnen nur die Optionen **compress** und **hdspace** zur Verfügung. Während die **hdspace** exakt das Gleiche meint, können Sie bei **compress** statt **on** auch direkt **zlib**, **lz4** und **zstd** als gewünschten Algorithmus angeben.

Da Sie nun ein Software-RAID mit ZFS oder btrfs verwenden, haben Sie auch mehrere Bootloader. Die erste einzelne Disk oder das erste RAID0, RAID1 oder RAID10 sowie jede Disk in einem RAID-Z erhält einen Bootloader. Der Server kann also selbst nach Ausfall einer Disk weiterhin über den Bootloader einer anderen Disk im RAID gestartet werden (aber natürlich nicht bei einer einzelnen Disk oder einer RAID0-Konfiguration).

Wenn Sie die richtigen System-Disks ausgewählt haben (siehe Abbildung 4.10), können Sie weiter zur nächsten Seite springen (siehe Abbildung 4.11). Wie schon gesagt, um die Datenspeicher für VMs und Container kümmern wir uns später.

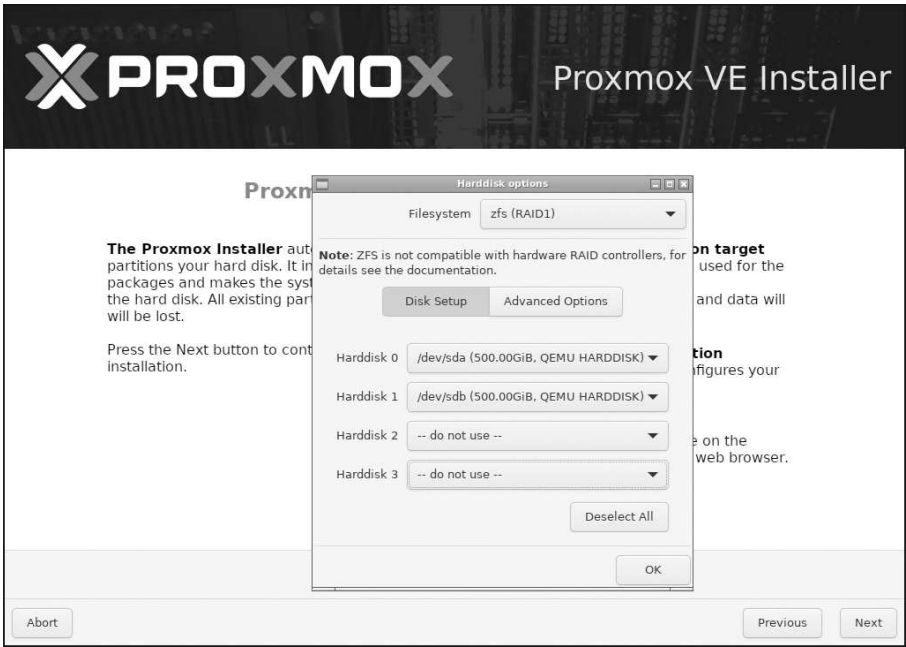


Abbildung 4.10: ZFS – RAID1 für die gewählten System-Disks

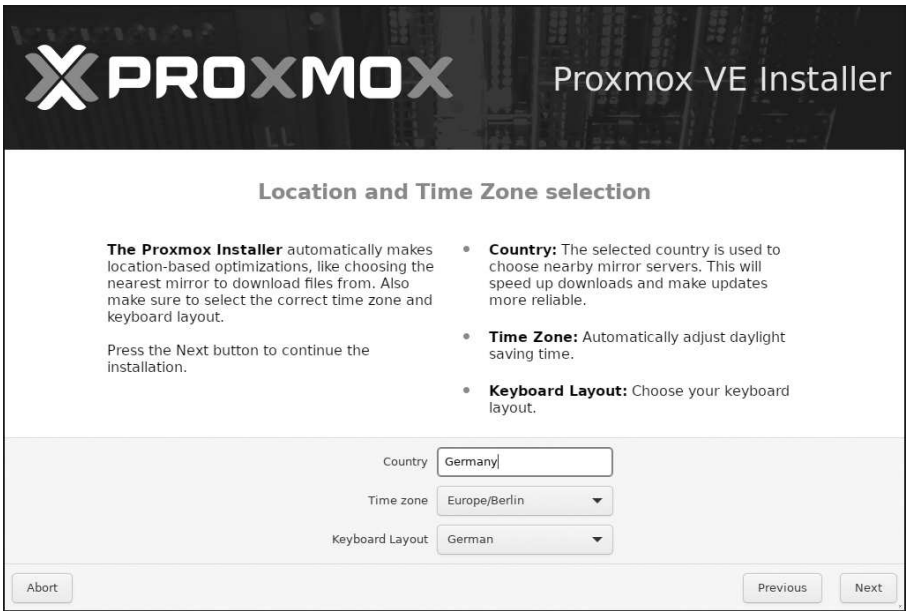


Abbildung 4.11: Seite 3 – Land, Zeitzone, Tastaturbelegung

Auf der darauffolgenden Seite legen Sie die Einstellungen für das Land, die Zeitzone sowie die gewünschte Tastaturbelegung fest.

Sofern beim Booten des Installers ein DHCP-Request erfolgreich war und der Server eine IP-Adresse erhalten hat, versucht der Installer, Ihren Aufenthaltsort über ein Traceroute an die Adresse 8.8.8.8 zu ermitteln. Ist dieser Schritt erfolgreich, sehen Sie ziemlich sicher das Land, in dem Sie sich befinden, die richtige Zeitzone sowie die richtige Tastaturbelegung.

Ist der Server nicht mit einem Netzwerk verbunden, erhalten Sie die von Proxmox festgelegten Voreinstellungen. Dies kann auch ein Hinweis darauf sein, dass Ihr Server noch keine Netzwerkverbindung aufbauen konnte.

Passen Sie die Einstellungen gerne nach Ihren Wünschen an.

PROXMOX Proxmox VE Installer

Administration Password and Email Address

Proxmox Virtual Environment is a full featured, highly secure GNU/Linux system, based on Debian.

In this step, please provide the *root* password.

- **Password:** Please use a strong password. It should be at least 8 characters long, and contain a combination of letters, numbers, and symbols.
- **Email:** Enter a valid email address. Your Proxmox VE server will send important alert notifications to this email account (such as backup failures, high availability events, etc.).

Press the Next button to continue the installation.

Password

Confirm

Email

Abbildung 4.12: Seite 4 – Passwort und E-Mail-Adresse des Administratorkontos

Nun ist es an der Zeit, das Kennwort für den **root**-User und die E-Mail-Adresse für Benachrichtigungen festzulegen (siehe Abbildung 4.12). Sie können beides später bei Bedarf jederzeit anpassen.

Die E-Mail-Adresse dient zur Benachrichtigung über den integrierten Postfix-Mailserver und kommt beispielsweise für abgeschlossene Backup-Jobs, bei Update-Benachrichtigungen oder bei (Teil-)Ausfällen eines Clusters zum Einsatz.

Ich hatte es bereits erwähnt: Beim Bootvorgang fordert der Server per DHCP eine IP-Adresse an. Erhält er eine, wird diese im Feld **IP-Address (CIDR)** als Voreinstellung gesetzt (siehe Abbildung 4.13). Allerdings ist es keine gute Idee, eine IP-Adresse aus dem DHCP-Bereich hier fest zu hinterlegen. Daher sollten Sie hier eine feste IP-Adresse außer-

halb Ihrer DHCP-Range eintragen oder diese IP-Adresse aus dem Pool Ihres DHCP entfernen oder als reserviert markieren.

PROXMOX Proxmox VE Installer

Management Network Configuration

Please verify the displayed network configuration. You will need a valid network configuration to access the management interface after installing.

After you have finished, press the Next button. You will be shown a list of the options that you chose during the previous steps.

- **IP address (CIDR):** Set the main IP address and netmask for your server in CIDR notation.
- **Gateway:** IP address of your gateway or firewall.
- **DNS Server:** IP address of your DNS server.

Management Interface: ens18 - bc:24:11:54:38:ab (virtio_net) ▼

Hostname (FQDN): pve.local

IP Address (CIDR): 10.1.99.187 / 24

Gateway: 10.1.99.254

DNS Server: 10.1.99.254

Buttons: Abort, Previous, Next

Abbildung 4.13: Seite 5 – Netzwerkeinstellungen

Das **Management Interface** setzt die Netzwerkschnittstelle, die mit der Standard-Bridge `vmbr0` verbunden wird und über die Ihr Proxmox-Server erreichbar ist und alle VMs und Container ebenfalls kommunizieren – zumindest so lange, bis Sie die Netzwerkkonfiguration ändern.

Die Einträge bei **Hostname**, **Gateway** und **DNS Server** passen Sie Ihrer Infrastruktur entsprechend an, damit der Server mit der Außenwelt kommunizieren kann – sofern Sie das wünschen. Ein Air-Gapped-Betrieb ist durchaus eine Option, und in der Praxis werden auch viele Hypervisoren so betrieben.

Auf der nächsten Seite erhalten Sie eine Zusammenfassung der von Ihnen gewählten Einstellungen (siehe Abbildung 4.14). Sie können den automatischen Neustart nach der Installation verhindern, indem Sie den Haken bei **Automatically reboot after successful installation** löschen.

Bis hierher können Sie jederzeit zurückgehen und Einstellungen korrigieren oder die Installation abbrechen.

Bestätigen Sie diese Seite und fahren Sie fort, werden die angegebenen Disks gelöscht und Proxmox VE wird installiert.

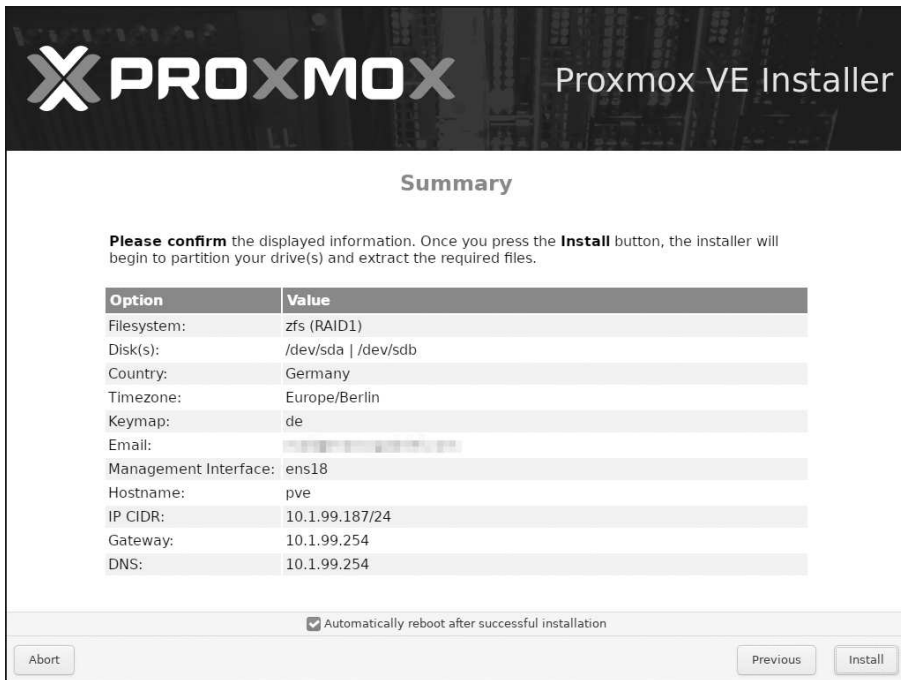


Abbildung 4.14: Seite 6 – Zusammenfassung

Wie in Abbildung 4.15 erscheinen ein Fortschrittsbalken sowie einige Informationen über Proxmox VE im oberen Teil des Fensters.

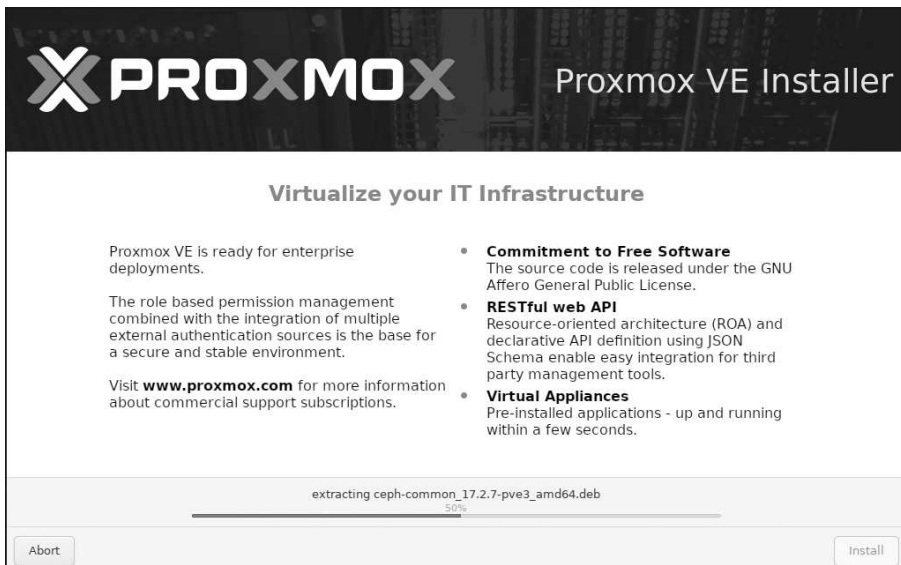


Abbildung 4.15: Die Installation läuft ...

Sollten Sie an dieser Stelle jemals ein Problem bekommen oder einfach das Bedürfnis haben, ein wenig mehr über den Installationsvorgang zu erfahren, dann folgen Sie mir bitte hier entlang: Mit einigen Tastenkombinationen können Sie auf andere Terminals umschalten und weitere Informationen sehen.

```
initial setup startup
mounting proc filesystem
mounting sys filesystem
boot comandline: BOOT_IMAGE=/boot/linux26 ro ramdisk_size=16777216 rw quiet spla
loading drivers: i2c_piix4 pata_acpi vmgenid floppy qemu_fw_cfg mac_hid uhci_hc
pkr intel_cstate rapl aesni_intel sha1_ssse3 sha256_ssse3 ghash_clmulni_intel po
intel intel_rapl_common
modprobe: ERROR: could not insert 'intel_cstate': No such device
searching for block device containing the ISO proxmox-ve-8.3-1
with ISO ID '20c7412e-a788-11ef-9ee4-4f61d2460cfb'
testing device '/dev/sr0' for ISO
found Proxmox VE ISO
preparing installer mount points and working environment
switching root from initrd to actual installation system
Starting Proxmox installation
Installing additional hardware drivers
Starting hotplug events dispatcher: systemd-udevd.
Synthesizing the initial hotplug events (subsystems)...done.
Synthesizing the initial hotplug events (devices)...done.
Waiting for /dev to be fully populated...done.
mount: devpts mounted on /dev/pts.
/bin/dbus-daemon
starting D-Bus daemon
Attempting to get DHCP leases... Internet Systems Consortium DHCP Client 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/ens19/bc:24:11:0f:e6:0c
Sending on LPF/ens19/bc:24:11:0f:e6:0c
Listening on LPF/ens18/bc:24:11:54:38:ab
Sending on LPF/ens18/bc:24:11:54:38:ab
Sending on Socket/fallback
DHCPDISCOVER on ens19 to 255.255.255.255 port 67 interval 3
DHCPDISCOVER on ens18 to 255.255.255.255 port 67 interval 8
DHCPOFFER of 10.1.99.126 from 10.1.99.252
DHCPREQUEST for 10.1.99.126 on ens19 to 255.255.255.255 port 67
DHCPOFFER of 10.1.99.187 from 10.1.99.252
DHCPREQUEST for 10.1.99.187 on ens18 to 255.255.255.255 port 67
DHCPACK of 10.1.99.126 from 10.1.99.252
bound to 10.1.99.126 -- renewal in 220 seconds.
done
Starting chrony for opportunistic time-sync...
Starting a root shell on tty3.
trying to detect country...
detected country: DE
Starting the installer GUI - see tty2 (CTRL+ALT+F2) for any errors...
```

Abbildung 4.16: **[Strg] + [Alt] + [F1]** – Bootscreen

Den Bootscreen erreichen Sie über die Tastenkombination **[Strg] + [Alt] + [F1]** (siehe Abbildung 4.16). Hier sehen Sie, ob der Installer normal gestartet ist und ob der DHCP-Request erfolgreich war und welche IP-Adresse dem Server zugeteilt wurde.

Sollte die Installation an irgendeiner Stelle einmal abbrechen, könnten Sie mit den Tasten **Strg** + **Alt** + **F2** zum Installer-Log umschalten und sehen, ob es gerade ein Problem bei einem bestimmten Paket gibt (siehe Abbildung 4.17).

```
(Reading database ... 19363 files and directories currently installed.)
Preparing to unpack .../libnet-subnet-perl_1.03-2_all.deb ...
Unpacking libnet-subnet-perl (1.03-2) ...
Selecting previously unselected package libnet1:amd64.
(Reading database ... 19369 files and directories currently installed.)
Preparing to unpack .../libnet1_1.1.6+dfsg-3.2_amd64.deb ...
Unpacking libnet1:amd64 (1.1.6+dfsg-3.2) ...
Selecting previously unselected package libnetaddr-ip-perl.
(Reading database ... 19378 files and directories currently installed.)
Preparing to unpack .../libnetaddr-ip-perl_4.079+dfsg-2+b1_amd64.deb ...
Unpacking libnetaddr-ip-perl (4.079+dfsg-2+b1) ...
Selecting previously unselected package libnetfilter-conntrack3:amd64.
(Reading database ... 19461 files and directories currently installed.)
Preparing to unpack .../libnetfilter-conntrack3_1.0.9-3_amd64.deb ...
Unpacking libnetfilter-conntrack3:amd64 (1.0.9-3) ...
Selecting previously unselected package libnetfilter-log1:amd64.
(Reading database ... 19466 files and directories currently installed.)
Preparing to unpack .../libnetfilter-log1_1.0.2-3_amd64.deb ...
Unpacking libnetfilter-log1:amd64 (1.0.2-3) ...
Selecting previously unselected package libnft0.52:amd64.
(Reading database ... 19473 files and directories currently installed.)
Preparing to unpack .../libnft0.52_0.52.23-1+b1_amd64.deb ...
Unpacking libnft0.52:amd64 (0.52.23-1+b1) ...
Selecting previously unselected package libnftnl0:amd64.
(Reading database ... 19565 files and directories currently installed.)
Preparing to unpack .../libnftnl0_1.0.2-2_amd64.deb ...
Unpacking libnftnl0:amd64 (1.0.2-2) ...
Selecting previously unselected package libnfsidmap1:amd64.
(Reading database ... 19570 files and directories currently installed.)
Preparing to unpack .../libnfsidmap1_2.6.2-4_amd64.deb ...
Unpacking libnfsidmap1:amd64 (1:2.6.2-4) ...
Selecting previously unselected package libnftables1:amd64.
(Reading database ... 19581 files and directories currently installed.)
Preparing to unpack .../libnftables1_1.0.6-2+deb12u2_amd64.deb ...
Unpacking libnftables1:amd64 (1.0.6-2+deb12u2) ...
Selecting previously unselected package libnftnl11:amd64.
(Reading database ... 19588 files and directories currently installed.)
Preparing to unpack .../libnftnl11_1.2.4-2_amd64.deb ...
Unpacking libnftnl11:amd64 (1.2.4-2) ...
Selecting previously unselected package libnl-3-200:amd64.
(Reading database ... 19593 files and directories currently installed.)
Preparing to unpack .../libnl-3-200_3.7.0-0.2+b1_amd64.deb ...
Unpacking libnl-3-200:amd64 (3.7.0-0.2+b1) ...
Selecting previously unselected package libnl-route-3-200:amd64.
(Reading database ... 19604 files and directories currently installed.)
Preparing to unpack .../libnl-route-3-200_3.7.0-0.2+b1_amd64.deb ...
Unpacking libnl-route-3-200:amd64 (3.7.0-0.2+b1) ...
```

Abbildung 4.17: **Strg** + **Alt** + **F2** – Installer-Log

Sie möchten sich auf dem frisch installierten System schon einmal umschauen, Administrationstätigkeiten oder gar Änderungen vornehmen? Dann erreichen Sie über **Strg** + **Alt** + **F3** eine frische **root**-Shell, mit der Sie bereits vollen Zugriff während der Installation haben (siehe Abbildung 4.18). Bedenken Sie, dass Sie sich im System des laufenden Installers befinden und noch nicht auf dem fertig installierten Proxmox-Hypervisor.

```
Debian GNU/Linux 12 proxmox tty3
proxmox login: root (automatic login)

Linux proxmox 6.8.12-4-pve #1 SMP PREEMPT_DYNAMIC PMX 6.8.12-4 (2024-11-06T15:04Z) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@proxmox:/# _
```

Abbildung 4.18: **[Strg] + [Alt] + [F3]** – eine frische root-Konsole erwartet Sie.

Diese root-Shell kann sehr praktisch sein, wenn Sie vor der Installation über die Konsole schon VLANs oder Storages einbinden möchten.

Sie haben sich umgesehen, möchten nun aber sehen, wie weit der Installer fortgeschritten ist? Kehren Sie mit **[Strg] + [Alt] + [F4]** zum grafischen Installer zurück (siehe Abbildung 4.19).



Abbildung 4.19: **[Strg] + [Alt] + [F4]** – zurück zum grafischen Installer

Am Ende der Installation rebootet der Server automatisch und startet den grade installierten Proxmox-VE-Hypervisor (siehe Abbildung 4.20).

Sie werden von einem freundlichen Terminal empfangen. Netterweise erhalten Sie oben den Hinweis, wie Sie sich über einen Browser nun mit der GUI verbinden können.

Damit ist die Installation abgeschlossen und Ihr Hypervisor ist einsatzbereit.

```

Initial setup startup
mounting proc filesystem
mounting sys filesystem
boot commandline: BOOT_IMAGE=/boot/linux26 ro ramdisk_size=16777216 rw quiet spla
loading drivers: i2c_piix4 pata_acpi vmgenid floppy qemu_fw_cfg mac_hid uhci_hc
bkr intel_cstate rapl aesni_intel sha1_ssse3 sha256_ssse3 ghash_clmulni_intel po
intel_intel_rapl_common
modprobe: ERROR: could not insert 'intel_cstate': No such device
searching for block device containing the ISO proxmox-ve-8.3-1
with ISO ID '20c7412e-a788-11ef-9ee4-4f61d2460cfb'
testing device '/dev/sr0' for ISO
found Proxmox VE ISO
preparing installer mount points and working environment
switching root from initrd to actual installation system
Starting Proxmox installation
Installing additional hardware drivers
Starting hotplug events dispatcher: systemd-udevd.
Synthesizing the initial hotplug events (subsystems)...done.
Synthesizing the initial hotplug events (devices)...done.
Waiting for /dev to be fully populated...done.
mount: devpts mounted on /dev/pts.
/bin/dbus-daemon
starting D-Bus daemon
Attempting to get DHCP leases... Internet Systems Consortium DHCP Client 4.4.3-P1
Copyright 2004-2022 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/ens19/bc:24:11:0f:e6:0c
Sending on LPF/ens19/bc:24:11:0f:e6:0c
Listening on LPF/ens18/bc:24:11:54:38:ab
Sending on LPF/ens18/bc:24:11:54:38:ab
Sending on Socket/fallback
DHCPDISCOVER on ens19 to 255.255.255.255 port 67 interval 3
DHCPDISCOVER on ens18 to 255.255.255.255 port 67 interval 8
DHCPOFFER of 10.1.99.126 from 10.1.99.252
DHCPREQUEST for 10.1.99.126 on ens19 to 255.255.255.255 port 67
DHCPOFFER of 10.1.99.187 from 10.1.99.252
DHCPREQUEST for 10.1.99.187 on ens18 to 255.255.255.255 port 67
DHCPACK of 10.1.99.126 from 10.1.99.252
bound to 10.1.99.126 -- renewal in 220 seconds.
done
Starting chrony for opportunistic time-sync...
Starting a root shell on tty3.
trying to detect country...
detected country: DE
Starting the installer GUI - see tty2 (CTRL+ALT+F2) for any errors...

```

Abbildung 4.20: Nach Installation und Neustart landen Sie auf der Konsole und es kann losgehen.

4.1.3 Installation von Proxmox VE auf Basis eines Debian Linux

Eine zweite Möglichkeit ist die Installation von Proxmox VE auf einem Debian Linux in der 64-Bit-Variante (amd64). Diesen Weg sollten Sie wählen, wenn Sie besondere Anforderungen haben, die der offizielle Proxmox-Installer nicht bereitstellt. Der Support dieser Variante ist ebenfalls durch den Hersteller gewährleistet.

Entscheiden Sie sich für diese Variante, wenn Sie beispielsweise

- eine vollständige Verschlüsselung (inklusive der root-Partition) mit LUKS verwenden möchten,
- während der Installation bereits Zugriff auf ein Netzwerk in einem VLAN brauchen (wobei dies vielleicht auch über die root-Konsole des Installers umsetzbar wäre).

- auf ein besonderes Partitionsschema setzen wollen und/oder
- ein root-Filesystem einsetzen möchten, das der PVE-Installer nicht zur Auswahl stellt.

Die Installation einer bestimmten Proxmox-VE-Version ist nur auf einer bestimmten Version von Debian möglich. Ein Proxmox VE 9.x setzt Debian 13 (Trixie) voraus, Version 8.x benötigte Debian 12 (Bookworm), und ich wage die Voraussage, dass PVE in der Version 10.x ein Debian 14 (Forky) voraussetzen wird.

Nicht vom Hersteller unterstützt wird übrigens die Installation auf

- einem 32-Bit Linux,
- einer anderen Version von Debian Linux,
- einem anderen Linux-System als Debian (selbst dann nicht, wenn es sich um Debian-Derivate wie Ubuntu oder Devuan Linux handelt) und/oder
- die Nutzung eines Linux-Software-RAID (`mdraid`).

Sofern dies Ihren Plänen nicht im Wege steht, können Sie nun mit der Installation beginnen.

Installation eines Debian 13 (Trixie) als Basissystem

Starten Sie, indem Sie Debian 13 (Trixie) installieren. Folgen Sie dem Installer, bis das Debian-System vollständig und nach Ihren Wünschen installiert ist.

Verwenden Sie eine statische IPv4-Adresse bei der Einrichtung des Netzwerks und stellen Sie sicher, dass der Hostname des Systems auf genau diese IP-Adresse aufgelöst wird. Normalerweise wird eine Abfrage des Hostnamens ja in die IPv4-Loopback-Adresse `127.0.0.1` oder die IPv6 Loopback-Adresse `::1` aufgelöst. Wenn Sie dies jedoch einfach so beibehalten, werden die Proxmox-Services nicht starten und Ihr Hypervisor wird nicht funktionsfähig sein.

Passen Sie daher mit einem Editor Ihrer Wahl die Datei `/etc/hosts` an, indem Sie die richtige Adresse, beispielsweise `10.10.1.1`, mit dem vollständigen Hostnamen inklusive Domain und Alias hinterlegen:

```
127.0.0.1 localhost
10.11.12.13 pve01.mydomain.com pve01 <- Hier die IP-Adresse, den FQDN
und den Alias hinterlegen

# The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Testen Sie die Einstellung anschließend und stellen Sie sicher, dass eine von außen erreichbare IP-Adresse als Ergebnis angezeigt wird und nicht die Loopback-Adresse:

```
hostname --ip-addresses
10.11.12.13
```

Bei der Paketauswahl genügt es, die **Standard System Utilities** und den **SSH server** auszuwählen. Alle anderen benötigten Pakete werden von Proxmox in eigenen, teilweise angepassten Versionen bereitgestellt und werden später aus den Proxmox-Repositories geladen.

Eine Desktop-Umgebung wie Gnome, KDE oder Xfce4 können Sie zwar installieren, Sie brauchen diese jedoch nicht. Ich empfehle auch, auf einem Hypervisor nur die Software zu installieren, die für den Betrieb benötigt wird. Dank der Web-GUI und dem SSH-Zugang gehört ein Desktop-Environment meist nicht dazu. Eine Ausnahme gibt es jedoch: Wenn Sie auch selbst für Proxmox entwickeln möchten, kann eine Desktop-Umgebung als Teil einer Developer-Workstation sinnvoll sein. Unter https://pve.proxmox.com/wiki/Developer_Workstations_with_Proxmox_VE_and_X11 finden Sie weitere Informationen zu diesem Thema.

Mehr Details zur Installation von Debian Linux als Grundlage erhalten Sie im Proxmox-Wiki unter https://pve.proxmox.com/wiki/Install_Proxmox_VE_on_Debian_13_Trixie.

Einbinden der Proxmox-VE-Repositories

Um die PVE-Pakete herunterzuladen und dann installieren zu können, teilen Sie dem Debian Linux zunächst mit, woher es diese Pakete erhalten kann. Sie müssen also die Paketquellen (*Repositories*) erweitern bzw. die von Proxmox zusätzlich hinterlegen:

```
~# cat > /etc/apt/sources.list.d/pve-install-repo.sources << EOL
Types: deb
URIs: http://download.proxmox.com/debian/pve
Suites: trixie
Components: pve-no-subscription
Signed-By: /usr/share/keyrings/proxmox-archive-keyring.gpg
EOL
```

Damit Debian das Repository als gültig anerkennt, wird noch der GPG-Schlüssel für es benötigt. Diesen laden Sie mit wenigen Schritten herunter und deklarieren ihn als vertrauenswürdig:

```
~# wget https://enterprise.proxmox.com/debian/proxmox-archive-keyring-trixie.gpg -O /usr/share/keyrings/proxmox-archive-keyring.gpg
```

Um ganz sicher zu gehen, dass Sie den richtigen Schlüssel importiert haben, verifizieren Sie ihn anschließend:

```
~# sha256sum /usr/share/keyrings/proxmox-archive-keyring.gpg
136673be77aba35dcce385b28737689ad64fd785a797e57897589aed08db6e45 /usr/
share/keyrings/proxmox-archive-keyring.gpg
```

```
~# md5sum /usr/share/keyrings/proxmox-archive-keyring.gpg
77c8b1166d15ce8350102ab1bca2fcfb /usr/share/keyrings/proxmox-archive-
keyring.gpg
```

Nun lesen Sie die verfügbaren Pakete aller Repositories neu ein und führen ein vollständiges Upgrade aller Pakete auf den aktuellen Stand durch:

```
~# apt update && apt full-upgrade
```

Installation des Proxmox-VE-Kernels

Dann führen Sie eine Installation des Proxmox-Kernels durch und starten den Server anschließend neu:

```
~# apt install proxmox-default-kernel
[...]
~# systemctl reboot
```

Installation der Proxmox-VE-Pakete

Nun installieren Sie die Proxmox-VE-Pakete nach, um aus dem Debian-Linux-Server einen Proxmox-VE-Hypervisor zu machen. Wundern Sie sich nicht darüber, dass viel mehr Pakete installiert werden als die paar hier angegebenen. Die Abhängigkeiten werden automatisch aufgelöst und installiert.

```
~# apt install proxmox-ve postfix open-iscsi chrony
```

Beantworten Sie die Fragen von `apt` bei der Installation entsprechend Ihrer Wünsche. Postfix sollte als *satellite system* konfiguriert werden, wenn Sie bereits einen Mailserver betreiben oder als *local only*, wenn Sie sich nicht sicher sind. Wie Benachrichtigungen zu Ihnen kommen, lässt sich auch später noch flexibel über das Notification-System von Proxmox konfigurieren.

Entfernen des Debian-Kernels

Da Sie nun den Proxmox-Kernel installiert haben und nach dem Reboot bereits nutzen, sollten Sie den ursprünglich mit Debian installierten Kernel entfernen:

```
# Kernel-Version daraufhin prüfen, ob die Version einen Hinweis
# auf "Proxmox" oder "pve" enthält
~# uname -a
Linux pve 6.14.11-1-pve #1 SMP PREEMPT_DYNAMIC PMX 6.14.11-1 (2025-08-
26T16:06Z) x86_64 GNU/Linux
# dann den Debian-Kernel entfernen
~# apt remove linux-image-amd64 'linux-image-6.12*'
```

Danach aktualisieren Sie den Bootloader des Systems:

```
~# update-grub
```

Abschluss der Installation

Ein erstes Login in der Web-GUI ist nun unter der Adresse <https://<ip-adresse>:8006> möglich. Wählen Sie die Option **PAM** bei **Realm** aus. Danach können Sie sich als Benutzer **root** und dem root-Passwort anmelden.

Ihr Hypervisor funktioniert nun zwar bereits, aber Ihre VMs oder Container könnten noch nicht mit der Außenwelt kommunizieren. Dazu müssen Sie noch eine Linux-Bridge erstellen. Wählen Sie für sie den Namen **vmbr0**, um möglichst gleiche Namen wie der ISO-Installer zu verwenden und eine weitere Bearbeitung durch die GUI zu ermöglichen. Haben Sie die **vmbr0**-Bridge erstellt, geben Sie ihr die IP-Adresse des physischen Interface und legen die Bridge auf das physische Interface. Um Ihnen dies zu erleichtern, zeige ich Ihnen hier eine Vorher-nachher-Beispielkonfiguration:

Ursprünglich sieht die Debian-Netzwerkconfiguration in der Datei `/etc/network/interfaces` so aus:

```
auto lo
iface lo inet loopback

auto ens18
iface ens18 inet static
    address 10.11.12.13/24
    gateway 10.11.12.1
```

Nach der Umstellung sollte sie so aussehen:

```
auto lo
iface lo inet loopback

iface ens18 inet manual # <- "static" auf "manual" umstellen,
                        "auto ens18" entfernen

auto vmbr0 # <- vmbr0 soll automatisch hochgefahren werden
iface vmbr0 inet static # <- Konfiguration von vmbr0 ab hier
    address 10.11.12.13/24
    gateway 10.11.12.1
    bridge-ports ens18 # <- vmbr0 wird mit der physischen ens18
                        verbunden
    bridge-stp off
    bridge-fd 0
```

Als letzten Schritt sollten Sie nun Ihre Proxmox-Subscription über das Web-Interface hochladen oder auf der Shell mit `pve-subscription set <subscription-key>` aktivieren. Entfernen Sie nun das Repository `no-subscription`, damit Sie die Updates aus dem Enterprise-Repository erhalten:

```
~# rm /etc/apt/sources.list.d/pve-install-repo.sources
```

Wenn in Ihrem Bootmenü später auch Einträge Ihrer virtuellen Maschinen auftauchen, sollten Sie das Paket `os-prober` von Debian deinstallieren. Mit dem Befehl `apt remove os-prober` ist das schnell erledigt. Nach einem weiteren `update-grub` sollten die zusätzlichen Einträge im Bootmenü verschwunden sein.

Die Installation ist nun vollständig abgeschlossen und Sie können Ihren neuen Proxmox-VE-Hypervisor nun verwenden. Hurra!

4.1.4 Automatische Installation

Nachdem Sie die Standardinstallation per ISO-Image und die manuelle Installation über ein Debian Linux kennengelernt haben, möchte ich Sie noch auf die Möglichkeit einer *unattended*, also einer unbeaufsichtigten, automatischen Installation von Proxmox VE hinweisen.

Mit Proxmox VE 8.0 hielt diese Funktion Einzug in das offizielle ISO. In älteren Versionen war eine automatische Installation zwar bereits möglich, aber komplizierter umzusetzen.

Die automatische Installation eines Debian Linux ist seit vielen Jahren über eine `preseed`-Datei möglich, aber nicht trivial. Alternativ dazu war es möglich, den Installer von Proxmox VE anzupassen, da er auch im Quellcode unter der AGPLv3-Open-Source-Lizenz frei unter <https://git.proxmox.com/?p=pve-installer.git;a=summary> verfügbar ist, und hier Ihr gewünschtes Installationsverfahren zu automatisieren.

Eine automatische, unbeaufsichtigte Installation von Proxmox VE ist meist für zwei Anwendungsfälle interessant:

- Sie müssen regelmäßig neue Server installieren, beispielsweise in einem Datacenter oder als Hardwarelieferant.
- Sie möchten absolut sichergehen, auf allen Servern die exakt gleiche Konfiguration anzuwenden.

Es gibt weitere Szenarien, die jedoch nicht ganz so gängig sind. Ich nutze den automatischen Installer mittlerweile, um Proxmox-Server für Trainings vorzubereiten, und sicherlich gibt es noch weitere valide Anwendungsmöglichkeiten.

Die automatische Installation basiert auf einer Antwortdatei (z. B. `answer.toml`), die alle notwendigen Installationsparameter enthält, wie z. B. das Passwort für den root-User sowie die Netzwerkconfiguration und Festplatteneinstellungen. Diese Datei kann Filterregeln ent-

halten, um die Installation auf spezifischen Geräten anhand von Eigenschaften wie Seriennummern oder MAC-Adressen durchzuführen.

Vorbereitung des Installationsmediums

Um die automatische Installation nutzen zu können, müssen Sie ein Installations-ISO-Image vorbereiten, das die Antwortdatei enthält oder auf diese verweist. Sie benötigen dazu das Tool `proxmox-auto-install-assistant` sowie zur Erstellung des neuen ISO-Images `xorriso`. Beides können Sie zusammen mit dem folgenden Befehl auf einem Proxmox-VE-Server installieren:

```
~# apt install proxmox-auto-install-assistant xorriso
```

Mit dem Befehl `proxmox-auto-install-assistant prepare-iso` kann ein bestehendes Proxmox-ISO angepasst und für die automatische Installation vorbereitet werden.

Vorher benötigen Sie jedoch noch die passende Antwortdatei für Ihre geplante Installation. Diese Antwortdatei kann dem Installer auf verschiedene Weisen bereitgestellt werden.

Mit der Option `--fetch-from` präparieren Sie das ISO-Image so, dass die Antwortdatei von einer vorher definierten Quelle gelesen wird. Folgende Möglichkeiten stehen Ihnen zur Verfügung:

- **Integriert in das ISO-Image:** Die Antwortdatei wird direkt in das ISO-Image eingebunden und bei der automatischen Installation von dort geladen. Sie haben damit keine externen Abhängigkeiten, und die automatische Installation beginnt, sobald Sie das ISO-Image von einem USB-Stick oder einem CD/DVD-Medium booten.

```
~# proxmox-auto-install-assistant prepare-iso \
    proxmox-ve_8.3-1.iso --fetch-from iso \
    --answer-file answer.toml
```

- **Auf separater Partition:** Die Antwortdatei befindet sich auf einer separaten Partition, z. B. einem USB-Stick mit dem Label `proxmox-ais`. Sie brauchen also neben dem Installationsmedium auch noch ein zweites Medium, auf dem die Konfiguration bereitgestellt wird. Sie benötigen damit nur ein ISO-Image und pro unterschiedlicher Konfiguration einen separaten USB-Stick. Diese Variante ist vor allem dann praktisch, wenn Sie immer wechselnde Konfigurationen installieren möchten.

```
~# proxmox-auto-install-assistant \
    prepare-iso /pfad/zum/source.iso --fetch-from partition
```

- **Über HTTP:** Die Antwortdatei wird während der Installation über das Netzwerk von einem HTTP-Server abgerufen. Sie können die URL der Antwortdatei auch per DNS-TXT-Record oder DHCP-Option 250 bereitstellen. Dies ist die flexibelste Option, da Sie die Antwortdatei jederzeit auf einem Server ändern oder über ein Skript beim Abruf generieren könnten.

Inhalt

1	Was ist Proxmox?	21
1.1	Einführung in Proxmox VE: Features	21
1.2	Open Source	24
1.3	Development – selbst etwas zu Proxmox VE beitragen oder selbst entwickeln	25
1.4	Die Proxmox-VE-Dokumentation verbessern	26
1.5	Vorteile mit Proxmox VE	27
1.5.1	Kostenvorteile	27
1.5.2	Einfachere Lizenzierung	27
1.5.3	Vermeiden eines Vendor-Lock-ins	27
1.5.4	Einfache Administration	28
1.6	Hilfe erhalten	28
1.7	Projekthistorie	28
1.8	Was Sie schon wissen sollten und was Sie lernen werden	31
1.8.1	Was Sie schon wissen sollten	31
1.8.2	Was Sie lernen werden	32
2	Grundlagen der Virtualisierung	33
2.1	Vorteile durch den Einsatz von Virtualisierung	34
2.2	Was Sie auf einem einzelnen Proxmox-VE-Server tun können	35
2.3	Was Sie auf mehreren Proxmox-VE-Servern tun können	36
3	Architektur und Hardware	37
3.1	Architektur	37
3.1.1	Die Basis: Debian Linux mit einem Custom-Kernel von Proxmox ...	37
3.1.2	Proxmox-VE-Services	38
3.1.3	Sonstige relevante Linux-Services	46
3.2	Hardware	54
3.2.1	CPU	55
3.2.2	RAM	55

3.2.3	RAID-Controller oder HBA	57
3.2.4	System-Disk	59
3.2.5	Daten-Disks für VMs und Container	59
3.2.6	Netzwerkkarten	60
3.2.7	Switches	62
4	Installation und Updates	65
4.1	Installation	65
4.1.1	Systemvoraussetzungen	66
4.1.2	Installation über den ISO-Installer von Proxmox VE	67
4.1.3	Installation von Proxmox VE auf Basis eines Debian Linux	83
4.1.4	Automatische Installation	88
4.2	Updates und Upgrades	92
4.2.1	Quellen für Updates	92
4.2.2	Proxmox-Repositories	92
4.2.3	Installation von Updates	95
4.2.4	Major-Upgrades	98
4.2.5	Major-Upgrade von Proxmox VE 8 auf 9	99
4.3	Die Upgrade-Schritte im Detail	105
4.3.1	VMs und Container migrieren oder herunterfahren	105
4.3.2	Repository-Einträge anpassen	106
4.3.3	Paketlisten aktualisieren	108
4.3.4	Dist-Upgrade durchführen	108
4.3.5	Alte oder obsolet gewordene Pakete entfernen	110
4.4	Zusätzliche Schritte bei Cluster-Setups	110
4.5	Abschließende Arbeiten	110
5	Benutzerverwaltung und Rechtemanagement	111
5.1	Die grundlegende Funktionsweise der rollenbasierten Zugriffskontrolle	111
5.1.1	Benutzer	111
5.1.2	Tokens	112
5.1.3	Gruppen	113
5.1.4	Privilegien	113
5.1.5	Rollen	114
5.1.6	Objekte	115

5.1.7	Resource Pools	116
5.1.8	Vererbung	117
5.2	Kombination der Komponenten und Anwendungsbeispiele	118
5.2.1	Beispiel: Die Administration bestimmter VMs an Entwickler delegieren	118
5.2.2	Beispiel: Anwendern erlauben, ihre VMs selbst neu zu starten	121
5.3	Berechtigungen konfigurieren und administrieren	122
5.3.1	Authentifizierung und Domänen (Realms)	122
5.3.2	2FA – Zwei-Faktor-Authentifizierung	132
6	Die Firewall in Proxmox VE	135
6.1	Features	135
6.2	Vererbung von Regeln und Definitionen	136
6.3	Aliases und IPSets	138
6.4	Vordefinierte Ausnahmen im lokalen Netzwerk	139
6.5	Firewall-Regeln definieren	141
6.6	Automatisch gesetzte Firewall-Ausnahmen	143
6.7	Security Groups	144
6.8	Logging von Firewall-Verbindungen	144
6.9	Aktivierung der Firewall	145
6.10	Installation der proxmox-firewall	146
6.11	Beispiele für Firewall-Regeln	147
7	Backup und Restore mit dem integrierten vzbump	149
7.1	Überblick und Architektur	149
7.2	Features und Einschränkungen des integrierten vzbump-Backups	149
7.3	Live-Snapshots und VMA-Archive	150
7.4	Backup-Fleecing	153
7.5	Backup-Modus	155
7.6	Backup-Ziele, Dateiformate und Kompression	156
7.7	Retention (Haltezeiten)	158

7.8	Backups durchführen	159
7.8.1	Manuelle Backups (ad hoc) erzeugen	160
7.8.2	Geplante Backups mit Backup-Jobs	162
7.9	Wiederherstellung gesicherter VMs und Container	165
7.9.1	Restore eines Backups und Überschreiben der Original-VM/ des Original-Containers	165
7.9.2	Restore eines Backups auf eine neue VM oder einen neuen Container	166
7.9.3	Restore von KVM-VMs per CLI mit qmrestore	167
7.9.4	Restore von Containern per CLI mit pct restore	168
7.10	Backup-Voreinstellungen pro Node in /etc/vzdump.conf	168
7.11	Best Practices für vzdump-Backups	169
8	Administration der Nodes	173
8.1	CPU-Microcode-Updates	173
8.1.1	Einrichtung der Microcode-Firmware beim Bootvorgang	174
8.1.2	Prüfen der bekannten CPU-Sicherheitslücken und deren Mitigation	174
8.2	Disk-Management	175
8.2.1	SMART-Werte mit smartctl auslesen	176
8.2.2	SMART bei Verwendung von RAID-Controllern	178
8.2.3	NVME-Devices mit nvme-cli beobachten	178
8.3	Node-Management mit pvenode	180
8.4	Zertifikate in Proxmox VE	181
8.4.1	ACME-Protokoll für automatisierte Zertifikate	183
8.4.2	Einbinden von Proxmox in eine eigene Certificate Authority (CA)	187
8.5	Task-Logs auf den Nodes	188
8.5.1	Funktionsweise des Task-Logs	188
8.5.2	Anzeige und Nutzung der Task-Logs in der GUI	189
8.5.3	Task-Logs über die CLI anzeigen und filtern	190
8.6	Node-Bulk-Actions	191
8.6.1	Alle Gäste mit pvenode startall starten	191
8.6.2	Alle Gäste mit pvenode stopall stoppen	192
8.6.3	Alle Gäste migrieren mit pvenode migrateall	192
8.7	Nodes mit Wake on LAN starten	193

8.8	Optimierung und Best Practices	194
8.8.1	Das Swapping anpassen	194
8.8.2	Die Anzahl der Worker-Threads konfigurieren	195
8.9	Sicherheitshärtung eines Proxmox-VE-Servers	196
8.10	Speicherdeduplizierung mit Kernel Samepage Merging (KSM)	198
8.10.1	Die KSM-Konfiguration anpassen	199
8.10.2	Security-Aspekte von KSM	200
8.10.3	KSM deaktivieren	201
9	Netzwerkconfiguration für Single-Server-Umgebungen	203
9.1	Networking-Support durch den Linux-Kernel	203
9.2	Netzwerkconfiguration per GUI und /etc/network/interfaces	203
9.2.1	Konfiguration des Netzwerks in der GUI	204
9.2.2	Konfiguration mit einem Editor in der /etc/network/interfaces	205
9.2.3	Anwenden der neuen Konfiguration	205
9.3	Netzwerktypen in Proxmox VE	206
9.4	Networking mit Linux-Netzwerkkomponenten	206
9.4.1	Linux-Bridges	206
9.4.2	Linux-Bonds	209
9.4.3	Linux-VLAN	214
9.5	Networking mit dem Open vSwitch (OVS)	218
9.5.1	Wo Open vSwitch seine Stärken ausspielt	218
9.5.2	Einschränkungen durch Open vSwitch	218
9.5.3	Installation der Open-vSwitch-Pakete	219
9.5.4	Die Komponenten von Open vSwitch	220
9.5.5	Konfigurationsbeispiele	220
9.6	Networking mit dem Software Defined Networking (SDN)	223
10	Storage in Proxmox VE	225
10.1	Wie Proxmox Storage verwendet	225
10.2	File-Storage	225
10.3	Block-Storage	226

10.4	Storage-Typen	226
10.4.1	Standard-Storage-Plugins	226
10.4.2	Unterschiede und Entscheidungshilfen	228
10.5	Lokaler Storage für einzelne Server	229
10.6	Shared Storage für Cluster-Umgebungen	231
10.6.1	Zentralisierter Shared Storage: NFS, iSCSI und FibreChannel	232
10.6.2	Multipathing für iSCSI- und FC-Storages	242
10.6.3	Distributed Shared Storage: Hyperkonvergenz mit Ceph oder Gluster	249
10.7	Storage-Leistung messen und optimieren	251
10.7.1	Durchsatz (Throughput)	251
10.7.2	IOPS (Input/Output Operations Per Second)	252
10.7.3	Latenz (Latency)	253
10.7.4	Das Zusammenspiel der Kennzahlen	253
10.7.5	Performance testen	255
10.8	ZFS-Storage	261
10.8.1	ZFS-Grundlagen	261
10.8.2	Anforderungen für den Betrieb von ZFS	262
10.8.3	Administration	263
10.8.4	ZFS-Troubleshooting und Fehlerbehebung	271
11	Virtuelle Maschinen mit KVM	279
11.1	KVM als Hypervisor	279
11.2	QEMU als Basis und Management	280
11.3	VirtIO – paravirtualisierte Treiber für bessere Leistung	281
11.3.1	Die gängigsten VirtIO-Komponenten	281
11.3.2	VirtIO unter Linux	282
11.3.3	VirtIO unter Windows	282
11.3.4	VirtIO-Devices in sonstigen Betriebssystemen	282
11.4	QEMU Guest Agent	283
11.4.1	Vorteile durch den Einsatz des QEMU Guest Agents	283
11.4.2	Installation und Konfiguration	285
11.4.3	Den Guest Agent für eigene Zwecke nutzen	286
11.5	Eine virtuelle Maschine (KVM) anlegen	292
11.5.1	Allgemeine Einstellungen	293
11.5.2	Einstellungen zur Betriebssysteminstallation	295
11.5.3	Einstellungen zum System	296

11.5.4	Konfiguration virtueller Disks	301
11.5.5	Einstellungen zur virtuellen CPU	306
11.5.6	Extra CPU-Flags	312
11.5.7	Eigene CPU-Modelle definieren mit <code>cpu-models.conf</code>	314
11.5.8	RAM-Konfiguration und dynamisches Speichermanagement	316
11.5.9	Netzwerk	317
11.5.10	Bestätigung und Abschluss der Konfiguration	319
11.6	Arbeit mit virtuellen Maschinen (KVM)	320
11.6.1	Suspend und Hibernation	320
11.6.2	Per Kommandozeilen-Interface	321
11.6.3	Dynamisches Speichermanagement bei KVM-VMs (Memory Ballooning)	322
11.6.4	Folder-Sharing zwischen Hypervisor und Gast mit VirtIO-FS	323
11.6.5	Per Kommandozeilen-Interface	328
11.7	PCI- und USB-Geräte an VMs durchreichen	328
11.7.1	Anwendungsszenarien, Vor- und Nachteile	328
11.7.2	Mediated Devices	329
11.7.3	IOMMU als notwendige Grundlage	330
11.7.4	Resource-Mappings: Passthrough im Cluster und mit HA	336
11.7.5	Verwendung der Mappings in einer VM	341
11.7.6	Die Vorteile im Cluster-Betrieb	342
11.8	Grafikkarten in VMs verwenden	343
11.8.1	GPU mit PCI-(Express-)Passthrough	343
11.8.2	GPU als VirtIO-GPU einer VM bereitstellen	344
11.8.3	GPU als VirGL-GPU in Linux VMs	345
11.8.4	GPU als PCI-(Express-)Passthrough in einen LXC-Container	345
12	LXC-Container	347
12.1	Grundlagen der Container-Virtualisierung	347
12.1.1	Architektur und Isolationsmechanismen	348
12.1.2	Unprivilegierte und privilegierte Container	348
12.1.3	LXC im Vergleich mit Docker: System- gegen Anwendungs- virtualisierung	349
12.2	Das Proxmox Container Toolkit (PCT)	350
12.3	LXC-Images und Templates	350

- 12.4 Storage und Mount-Points in LXC-Containern** 352
 - 12.4.1 Das Root-Filesystem rootfs 352
 - 12.4.2 Zusätzliche Mount-Points mp[n] 353
- 12.5 Erstellung und Konfiguration von Containern** 354
 - 12.5.1 Erstellung eines LXC-Containers über die GUI 354
 - 12.5.2 Einen neuen LXC-Container per CLI erzeugen 366
- 12.6 Migration und Hochverfügbarkeit** 367
 - 12.6.1 Migration von Containern 368
- 12.7 Management eines LXC-Containers** 368
 - 12.7.1 Fernzugriff und Ausführung von Befehlen 368
- 13 Subscriptions** 371
 - 13.1 Grundlagen der Subscription** 371
 - 13.1.1 Unterschied zum lizenzbasierten Finanzierungsmodell 372
 - 13.2 Subscription-Level** 373
 - 13.3 Wahl der passenden Subscription** 374
 - 13.3.1 Welche Support-Leistungen benötigen Sie? 374
 - 13.3.2 Wie viele CPUs haben Ihre Server? 374
 - 13.4 Die Anzahl und Level der Subscriptions bestimmen** 375
 - 13.4.1 Subscriptions für Single-Server-Umgebungen 375
 - 13.4.2 Subscriptions für Cluster-Umgebungen 375
 - 13.5 Subscription-Management** 376
 - 13.5.1 Subscription auf einem Server hinterlegen 376
 - 13.5.2 Subscription eines Servers auslesen 377
- 14 Migration zu Proxmox VE** 379
 - 14.1 Planung der Migration** 379
 - 14.1.1 Bestandsaufnahme (Inventarisierung) 379
 - 14.1.2 Storage- und Netzwerk-Analyse 380
 - 14.1.3 Treiber für unterschiedliche Hypervisor-Umgebungen 380
 - 14.2 Migrationspfade und Werkzeuge** 382
 - 14.2.1 Migration von VMware vSphere (ESXi) 382
 - 14.2.2 Migration von Microsoft Hyper-V oder anderen Systemen 386

14.3	Nach der Migration: Die Pflichtschritte	389
14.3.1	Alte Gast-Tools entfernen	389
14.3.2	VirtIO-Treiber installieren	389
14.3.3	Hardware auf VirtIO umstellen	390
14.3.4	QEMU Guest Agent installieren und aktivieren	390
14.4	Weitere Migrationspfade	390
15	Cluster mit Proxmox VE	391
15.1	Einführung in den Cluster-Bau mit Proxmox VE	391
15.2	Das Proxmox Cluster File System pmxcfs	392
15.3	Cluster-Kommunikation mit der Corosync Cluster Engine	394
15.3.1	Dediziertes Netzwerk für Corosync	395
15.3.2	Redundanz mit mehreren Corosync-Links schaffen	396
15.3.3	Geringe Latenzen sicherstellen	396
15.3.4	Anforderungen an Bandbreite	396
15.4	Basics des Cluster-Baus: Das Quorum	397
15.5	GUI, CLI und Editoren als Werkzeuge	398
15.5.1	GUI	399
15.5.2	Kommandozeile (CLI): pvecm, der Proxmox VE Cluster Manager	399
15.5.3	Editoren	400
15.6	Best Practices	400
15.7	Erstellen eines Proxmox-VE-Clusters	402
15.7.1	Namensauflösung	402
15.7.2	Zeitabgleich mit NTP	404
15.7.3	Schritt 1: Erstellen des Clusters auf dem ersten Node	407
15.7.4	Schritt 2 (optional): Join-Informationen abrufen	411
15.7.5	Schritt 3: Weitere Nodes hinzufügen	412
15.8	Cluster-Administration in der Praxis	415
15.8.1	Einen Node sicher aus dem Cluster entfernen	415
15.9	Status des Clusters prüfen und monitoren	416
15.9.1	Der schnelle Cluster-Check mit pvecm status	416
15.9.2	Corosync-Netzwerke mit corosync-cfgtool prüfen	418
15.10	Das Problem mit 2-Node-Clustern und mögliche Lösungen	419
15.10.1	Manuelles Erzwingen eines Quorums in Notfallsituationen	419

- 15.10.2 Einen Cluster um ein Quorum-Device erweitern 420
 - 15.10.3 Ein QDevice entfernen 422
- 15.11 Konkurrierende Aktionen durch Locks verhindern 423**
 - 15.11.1 Locking im Allgemeinen und unter Proxmox VE 423
 - 15.11.2 Manuelles Entfernen von Locks (Unlock) 424
- 16 Migration im Live-Betrieb 425**
 - 16.1 Live-Migration von VMs und Containern 425**
 - 16.1.1 Funktionsweise und Unterschiede 425
 - 16.1.2 Voraussetzungen für die Migration 426
 - 16.1.3 Eine Migration durchführen 426
 - 16.2 Storage-Migration 427**
 - 16.2.1 Funktionsweise der Storage-Migration bei QEMU/KVM-VMs 428
 - 16.3 Live-Migration von VMs mit lokalen Disks 429**
 - 16.4 Automatische Optimierung: TRIM nach der Migration 430**
 - 16.5 Remote-Migration von VMs und Containern zu externen Proxmox-VE-Servern 431**
 - 16.5.1 Vorbereitung des Zielnodes 432
 - 16.5.2 Remote-Migration auf dem Quellnode starten 433
 - 16.6 Optimierungsmöglichkeiten für Enterprise-Umgebungen 433**
 - 16.6.1 Das dedizierte Migrationsnetzwerk 434
 - 16.6.2 Secure oder Insecure Migration? 434
 - 16.6.3 Bandbreitenlimitierung 435
- 17 Hyperkonvergenz mit Proxmox VE und Ceph 437**
 - 17.1 Was ist Ceph? 437**
 - 17.1.1 Features 437
 - 17.1.2 Object-Storage, Block-Storage und Filesystem in einem 438
 - 17.1.3 Unterschied zu SAN- und NAS-Systemen 439
 - 17.1.4 Das Konzept der Hyperkonvergenz (HCI) 440
 - 17.1.5 Die CRUSH-Map 442
 - 17.1.6 Failure-Domains 444
 - 17.1.7 Der CRUSH-Algorithmus 444
 - 17.2 Architektur und Komponenten 445**
 - 17.2.1 Ceph-Daemons 445

17.2.2	Logische Einheiten	450
17.3	Voraussetzungen und Sizing eines Proxmox-VE-HCI-Clusters	454
17.3.1	Netzwerkanforderungen	455
17.3.2	Hardware-Sizing	459
17.4	Installation und Initialisierung	464
17.4.1	Installation der Ceph-Pakete	464
17.4.2	Erstellen der Monitore und Manager	469
17.4.3	Einrichten der OSDs	472
17.5	Konfiguration von Pools und Storage	477
17.5.1	Erstellen eines Ceph-Pools	478
17.5.2	Einbindung externer Ceph-Pools als RBD-Storage in Proxmox VE	481
17.6	CephFS – das verteilte Dateisystem	482
17.6.1	Einsatzszenarien	483
17.6.2	Einrichtung	483
17.7	Wartung und Betrieb	488
17.7.1	systemd-Daemons	488
17.7.2	Administration und Module des Managers	490
17.7.3	Verwendung der OSD Global Flags	491
17.7.4	Umgang mit Disk-Ausfällen	493
17.7.5	Cluster-Erweiterung	494
17.8	Monitoring und Troubleshooting	494
17.8.1	Die wichtigsten CLI-Befehle	495
17.8.2	Häufige Probleme und Lösungen	496
17.9	Best Practices: Scrubbing-Einstellungen anpassen	498
18	Asynchrone Replikation mit der Proxmox Storage Replication	499
18.1	Funktionsweise der Storage-Replikation	500
18.2	Failover-Szenarien beim Einsatz der Storage-Replikation	501
18.3	Voraussetzungen für die Storage-Replikation	502
18.4	Einrichten der Storage-Replikation	502
18.4.1	Lokalen ZFS-Storage konfigurieren	502
18.4.2	Replikationsjob anlegen	503
18.4.3	Überwachung und Verwaltung der Replikation	503
18.4.4	Best Practices und Optimierungen	504

19	Hochverfügbarkeit mit Proxmox-VE-Clustern	505
19.1	Was bedeutet HA in Proxmox VE?	505
19.2	Voraussetzungen für einen stabilen HA-Betrieb	507
19.2.1	Mehrere Server in einem Management-Cluster	507
19.2.2	Shared Storage wie Ceph, NFS, iSCSI/FC	507
19.2.3	Redundanz und 24/7-fähige Komponenten	508
19.2.4	Latenz-Anforderungen von Corosync	508
19.2.5	NTP muss funktionieren	508
19.2.6	Einsatz eines QDevice bei 2- oder 4-Node-Clustern	509
19.3	Der HA-Stack in Proxmox VE	509
19.3.1	Der Proxmox VE Cluster Resource Manager	509
19.3.2	Der Proxmox VE Local Resource Manager	510
19.3.3	Zusammenspiel des CRM mit den LRMs	511
19.3.4	Watchdog	511
19.3.5	Hardware-Watchdog statt Linux-Softdog verwenden	512
19.4	Basics des Cluster-Baus: Fencing – der Schutz vor Datenkorruption	513
19.4.1	Weshalb wird Fencing in einem HA-Cluster gebraucht?	513
19.4.2	Fencing sorgt für die Wahrung der Datenkonsistenz	514
19.5	Konfiguration und Praxis	515
19.5.1	HA-Management per GUI	516
19.5.2	Den HA-Manager per CLI nutzen	517
19.5.3	Regeln zur Node- und Ressourcenaffinität	518
19.5.4	Cluster Resource Scheduler (CRS) – intelligente Platzierung	520
19.5.5	Umgang mit geplanten Wartungsarbeiten	522
19.6	Einen HA-Cluster deaktivieren	523
20	Mit Hook-Skripten auf Events reagieren	525
20.1	Hook-Skripte in Backups	525
20.1.1	Hook-Events für Backups	526
20.1.2	Beispiel für ein einfaches Backup-Hook-Skript	529
20.2	Hook-Skripte in VMs und Containern	529
20.2.1	VM- und Container-Hook-Events	530
20.2.2	Beispiel für ein einfaches VM- bzw. Container-Hook-Skript	530

21	Templates und Clones	533
21.1	VM- und Container-Templates	533
21.2	Ein VM-Template vorbereiten und erzeugen	534
21.2.1	Autarkie mit Full Clones	534
21.2.2	Speicherplatzoptimierte Linked Clones	535
21.3	LXC-Appliances als Container-Templates	536
21.4	cloud-init für extern bereitgestellte Images	537
21.5	Proxmox-VE-Backups als Templates nutzen	538
22	Automatisierung mit der Proxmox-API	539
22.1	Der RESTful-Ansatz	540
22.1.1	Alles ist eine Ressource	540
22.1.2	Aktionen auslösen mit HTTP-Verben	540
22.1.3	Zustandslosigkeit (Statelessness)	541
22.1.4	Wege der Authentifizierung	541
22.1.5	Datenaustausch via JSON	542
22.2	Architektur und Komponenten	542
22.3	Authentifizierung	543
22.3.1	Ticket-basierte Authentifizierung	543
22.3.2	API-Tokens	545
22.4	Werkzeuge für den API-Zugriff	547
22.4.1	pvesh für eine lokale Nutzung der API	547
22.4.2	curl: API-Zugriff über ein Netzwerk	548
22.4.3	Client-Bibliotheken: Das Rad nicht neu erfinden	549
22.5	Arbeiten mit der API: Konzepte und Stolpersteine	549
22.5.1	Daten lesen mit GET	549
22.5.2	Aktionen ausführen mit POST, PUT und DELETE	550
22.5.3	Synchrone und asynchrone Tasks	550
22.5.4	Status einer UPID überwachen	551
23	Backups mit dem Proxmox Backup Server (PBS)	555
23.1	Einführung und Architektur	555
23.1.1	Warum vdump nicht mehr ausreicht	555

23.1.2	Was der PBS besser macht	556
23.1.3	Kaskadierung mehrerer PBS	558
23.2	Planung und Installation	560
23.2.1	Installationsmethoden	560
23.2.2	Hardware-Anforderungen und -Sizing	562
23.3	Einrichtung und Konfiguration	564
23.3.1	Datastores verwalten	564
23.3.2	Benutzerverwaltung und Berechtigungen	565
23.4	Integration in Proxmox VE	568
23.4.1	Hinzufügen des PBS als Storage	568
23.4.2	Client-Side Encryption (Verschlüsselung)	570
23.5	Backup und Restore in der Praxis	570
23.5.1	Backup-Jobs in Proxmox VE	571
23.5.2	Wiederherstellung von Backups	571
23.6	Wartung und Pflege	572
23.6.1	Pruning – Entsorgung abgelaufener Backups	573
23.6.2	Garbage Collection	573
23.6.3	Verify-Jobs	575
23.7	Air-Gapped mit Offsite-Sync, Tape-Backup und S3-Anbindung	576
23.7.1	Remote-Sync – der digitale Bunker	576
23.7.2	Tape-Backup	576
23.8	Ruhig schlafen	577
24	Erweiterte Netzwerkkonfiguration für Enterprise- und Cluster-Umgebungen	579
24.1	Physisches Cluster-Netzwerkdesign	579
24.1.1	Dediziertes oder konvergentes Netzwerk?	580
24.1.2	Multi-Chassis Link Aggregation (MLAG/Stacking)	584
24.2	Stern- oder Mesh-Topologie?	588
24.2.1	Switch-basierte Stern-Topologie	588
24.2.2	Direkt verkabelte Mesh-Topologie	589
24.2.3	Implementierung der Mesh-Topologie	591
24.3	Segmentierung von VM-Netzwerken	603
24.3.1	VLANs an VM-Interfaces	603
24.3.2	VLANs auf dedizierten Bridges	604
24.3.3	Guest-Based VLAN-Tagging für Firewalls oder Router	605

24.3.4	Layer-3/4-Isolation durch Firewall-Regeln	606
24.3.5	Isolation aller Interfaces untereinander	607
25	Software-Defined Networking (SDN) in Proxmox VE	609
25.1	Grundlagen des Software-Defined Networkings	609
25.2	Integration in Proxmox VE	610
25.3	Anforderungen an die physische Netzwerkanbindung	611
25.4	Die Komponenten des Proxmox-VE-SDN: Zone, VNet und Subnet	613
25.4.1	Zonen	614
25.4.2	VNet	615
25.4.3	Subnet	616
25.5	Zone-Types und deren Technologien	616
25.5.1	Simple Zone	616
25.5.2	VLAN-Zone	618
25.5.3	QinQ-Zone	619
25.5.4	VXLAN-Zone	620
25.5.5	EVPN-Zone	622
25.6	Einsatz eines Routing-Controllers für EVPN-Zonen	623
25.6.1	Die Controller-Typen EVPN, BGP und IS-IS	623
25.6.2	Einrichtung des EVPN-Controllers	625
25.6.3	Routing und NAT	625
25.7	IP Address Management: IPAM und DHCP	626
25.7.1	IPAM-Plugins	626
25.7.2	Automatisches DHCP	627
25.8	Praktische Beispiele für den Einsatz des SDN	628
25.8.1	Szenario 1: Die isolierte Testumgebung	629
25.8.2	Szenario 2: Mandantenfähigkeit oder Multi-Tenant-Umgebung	629
25.8.3	Szenario 3: Die automatisierte CI/CD-Pipeline	630
	Index	631

Index

/etc/network/interfaces	203
/etc/vzdump.conf	168
24/7	508
3-2-1-Regel	169

A

Abonnementmodell	371
Access Control List (ACL)	118, 358
ACME	183
Active Directory (AD)	125, 127
Active-Backup-Bond	585
Adaptive Load Balancing (ALB)	211
Affinitätsregeln	518
Affinity Rules	518
AGPLv3-Lizenz	24, 27
Alignment Shift	267
AMD-V	55
API	
Architektur	542
Authentifizierung	543
Idempotenz	549
Komponenten	542
pvesh	547
RESTful	540
synchrone und asynchrone Tasks	550
Token	545
Werkzeuge	547
ARC-Cache	262
Architektur	37
Asciidoc	27
Ausfallsicherheit	243
Authentifizierung	122
Authentik	130
Automatisierung	539

B

Backup	149, 538
ad hoc	160
Air-Gapped	576
Architektur	149
Disk-to-Disk-to-Tape (D2D2T)	577
durchführen	159
Garbage Collection	573
geplantes	162
Hardware-Anforderungen	562

Hook-Skript	525
Kaskadierung	558
Live-Restores	571
manuelles	160
Modi	155
Planung	560
Pruning	573
Pull-Prinzip	559
Remote-Sync	576
Single File Restore	572
Sizing	562
Tape-Backup	576
Verify-Jobs	575
Voreinstellungen	168
wiederherstellen	165
Wiederherstellung	571
Ziel	156

Backup → Proxmox Backup Server (PBS)

Backup-Fleecing	153
Backups-Job	162
Ballooning-Device	322
Bandbreitenlimitierung	435
Base-Image	535
Benachrichtigungen	50
Benutzer	111
Berechtigungen	111
BGP-Controller	624
Bit Rot	261
Block	556
Block-Storage	226
BlueStore	449, 460
Bond	60, 209
Border Gateway Protocol (BGP)	614
Bridge	206
Bridged Network	206
btrfs	72, 229
Bucket	443

C

Calendar Events	42
Ceph	24, 250
Architektur	445
Betrieb	488
CLI-Befehle	495
Cluster erweitern	494
Cluster-Größe	454

Ceph (Forts.)	
CPU-Sizing	459
Deep Scrubbing	498
Diskausfall	493
Disks	461
externe Pools einbinden	481
Features	437
Hardware-Sizing	459
häufige Probleme	496
HBA vs. RAID	463
Installation	464
logische Einheiten	450
Manager	490
Manager anlegen	469
MDS (Metadata-Server)	450
MGR (Manager)	447
MON (Monitor)	445
Monitor anlegen	469
Monitoring	494
Netzwerkanforderungen	455
ODS anlegen	472
OSD (Object Storage Daemon)	448
OSD Global Flags	491
Partitionierung	451
PG Autoscaler	454
Placement Group	453
Placement Group Autoscaler	448
Pools konfigurieren	477
Pool-Typen	452
Public Network	456
Quorum	445, 446
RAM-Sizing	460
Scrubbing	498
Slow Requests	496
Storage konfigurieren	477
Troubleshooting	494
vs. SAN- und NAS-Systeme	439
Wartung	488
Ceph Monitor (MON)	445
ceph status	495
CephFS	482
einrichten	483
Metadata-Server	484
Certificate Authority (CA)	181, 187
cgroups	279, 347
Changed Block Tracking	556
Chrony	46
chrony	396
Chunk	556
Chunk-Level Deduplication	556
Clock Skew	404
Clones	533
Full Clones	533
Linked Clones	533
cloud-init	537
Cluster	391
Administration	415
bauen	391
Best Practices	400
erstellen	402
Join-Informationen	411
Kommunikation	394
Latenz	396
per CLI initialisieren	410
per GUI initialisieren	408
QDevice hinzufügen	420
Quorum	397
Quorum prüfen	416
Status prüfen	416
Cluster Resource Manager (CRM)	45, 509
Zusammenspiel mit LRMs	511
Cluster Resource Scheduler (CRS)	520
Funktionsweise	521
Recovery-Szenario	522
Cluster-Map	446
Cluster-Network	456
Community-Forum	28
Container	
Escape	348
per CLI wiederherstellen	168
Virtualisierung	347
Corosync	44, 508
Bandbreite	396
Links	396
Netzwerk	395, 582
Netzwerk prüfen	418
Corosync Cluster Engine	392, 394
corosync-cfgtool	418
CPU	55
CPU-Microcode	173
Updates	173
cpu-models.conf	314
CPU-Over-Provisioning	307
cron	42, 53
CRUD	540
CRUSH (Controlled Replication Under Scalable Hashing)	437, 442
CRUSH-Map	442
CSRFPreventionToken	541, 544
curl	548
Custom-Kernel	37

D

Data Plane Development Kit (DPDK)	218
Debian Linux	31, 37, 38
Deduplizierung	556
ZFS	263
Deep Scrubbing	498
Developer-Mailingliste	25
Device-Redundanz	440
DHCP	626, 627
Dirty Bitmaps	558
Dirty Map	556
Disk-Management	175
Disk-to-Disk-to-Tape (D2D2T)	577
Distributed Shared Storage	249
Dist-Upgrade	108
dnsmasq	627
Dokumentation	26, 31
Domäne	122
Durchsatz	251

E

End-User-License-Agreement (EULA)	69
Entra ID (früher Azure AD)	130
Equal-Cost-Multipathing (ECMP)	596, 602
Erase-Coded Pools	452
Ethernet Virtual Private Network (EVPN)	614, 622
Events	525
EVPN-Controller	624
Exit-Nodes	625

F

Fail Over	243
Fail2ban	198
Failure-Domain	440, 444
Fault Tolerance	505
Features	21
Fencing	513
Hardware-	514
Self-Fencing	515
FibreChannel	232
File-Storage	225
FIO (Flexible I/O Tester)	256
Firewall → Proxmox VE Firewall	
Firmware	298
Fleecing	154
Folder-Sharing	323

FRRouting Software Suite (FRR)	596, 625
Full Clone	534

G

Garbage Collection	573
GFS2	241
GlusterFS	250
GNU AGPLv3	371
Google Identity Platform	130
Gotify	50, 52
GPU	
<i>KI-Anwendungen</i>	344
<i>LXC-Container</i>	345
GPU (Graphics Processing Unit)	343
Grafikkarte	343
Gruppe	113
Guest-Based VLAN-Tagging	605

H

HA Node Affinity Rules	519
HA Resource Affinity Rules	520
HA-Cluster	
<i>deaktivieren</i>	523
<i>konfigurieren</i>	515
HA-Management	516
<i>per CLI</i>	517
<i>per GUI</i>	516
HA-Node-Affinitätsregeln	518
Hardware	54
Hardware Compatibility List (HCL)	54
Hardware-Fencing	514
Hardware-RAID	70
Härtung	196
HBA	57
Heartbeat	396
Hibernation Mode	320
High Availability	
<i>Shared Storage</i>	507
Hook-Events	526
<i>für VMs und Container</i>	530
Hook-Skript	525
<i>Backup-Beispiel</i>	529
<i>für VMs und Container</i>	530
<i>in VMs und Containern</i>	529
Host-based Access Control	39
Host-Only Network	206
HTTP-Verben	540
HugePages	279

Hyperkonvergente Infrastruktur (HCI)	250
Hyperkonvergenz (HCI)	24, 249, 437, 440

I

i440fx	297
Idempotenz	549
IEEE 802.3ad	211
Immutable Storage	558
Incremental Forever	557
Initiator-Target-Mode (IT-Mode)	263
Installation	65
Instantiated Services	489
Inventarisierung	379
IOMMU	330
IOPS (Input/Output Operations Per Second)	252
IPAM	626
IPAM-Plugin	614
IPsec	621
IPSets	138
iSCSI	232
<i>Grundlagen</i>	233
<i>Initiator</i>	233
<i>Target</i>	233
<i>Verbindung konfigurieren</i>	233
iSCSI Qualified Name (IQN)	233
IS-IS-Controller	624
ISO-Image	65
IT-Mode	263

J

journalld	52
Jumbo-Frames	458

K

Kernel Samepage Merging (KSM)	53, 198
Kernel-based Virtual Machine → KVM	
Keycloak	130
KronosNet	395
ksmtuned	53
KVM	279

L

LACP	585
Lastverteilung	243
Latenz	253
LDAP	124

Leaky Bucket	306
Least-Privilege	118
Let's Encrypt	181
Lightweight Directory Access Protocol (LDAP)	124, 127
Link Aggregation Control Protocol (LACP)....	211
Linked Clone	535
Linux-Bridge	206
Linux-VLAN	214
Live-Migration	425
<i>durchführen</i>	426
<i>VMs mit lokalen Disks</i>	429
<i>Voraussetzungen</i>	426
Live-Snapshots	150
Lizenz	27
Load Balancing	209, 243
Local Resource Manager (LRM)	510
Lock	423
<i>entfernen</i>	424
Locks	196
Log-Analyse	52
Logical Unit Number (LUN)	232
Logical Volume Manager (LVM)	238
LVM	237
LXC-Appliances	536
LXC-Container	345, 347
<i>Architektur</i>	348
<i>erstellen per CLI</i>	366
<i>erstellen per GUI</i>	354
<i>Fernzugriff</i>	368
<i>GPU</i>	345
<i>managen</i>	368
<i>Migration</i>	368
<i>Mount-Points</i>	352
<i>privilegierte</i>	348
<i>rootfs</i>	352
<i>Storage</i>	352
<i>Templates</i>	350
<i>unprivilegierte</i>	348
<i>vs. Docker</i>	349

M

Maintenance-Mode	110
Major-Upgrades	98
Maschinentyp	297
MDevs	329
mdraid	66
Mediated Devices	329
Meltdown	312
Memory Ballooning	200, 322

Mesh-Topologie	589
<i>Broadcast-Bond</i>	592
<i>dynamisches Routing</i>	596
<i>implementieren</i>	591
<i>mit OpenFabric per GUI einrichten</i>	597
<i>statische Host-Routen</i>	593
Microsoft Hyper-V	33
Microsoft Standard Secure Boot Keys	299
Migration	379
<i>Bandbreitenlimitierung</i>	435
<i>Hyper V</i>	386
<i>Live</i>	425
<i>OVF/OVA</i>	384
<i>Pfade</i>	382
<i>planen</i>	379
<i>Restart</i>	425
<i>secure vs. insecure</i>	434
<i>Storage</i>	427
<i>Treiber</i>	380
<i>TRIM</i>	430
<i>VirtIO-Treiber</i>	389
<i>VMware vSphere</i>	382
<i>Vorabanalyse</i>	380
<i>Werkzeuge</i>	382
Migrationsnetzwerk	434
Mount-Points	353
Multi-Chassis Link Aggregation (MLAG)	586
<i>(MLAG/Stacking)</i>	584
Multi-Master-Design	22
Multipathing	242, 243

N

Namensauflösung	402
NAT Network	206
NetBox	626
Netfilter	135
Network File System (NFS)	232
Network Time Protocol	404
Netzwerkkarte	60
Netzwerkconfiguration	
<i>anwenden</i>	205
<i>Editor</i>	205
<i>GUI</i>	204
<i>Linux-Bond</i>	209
<i>Single-Server-Umgebung</i>	203
NFS	232
nftables	43, 135
Node entfernen	415
Node-Affinität	518

Node-Bulk-Actions	191
Node-Reboot	522
Noisy Neighbor	306
Notification Targets	50
NTP	46, 404, 508
NVIDIA GRID	344
nvme-cli	178

O

Objekt (RBAC)	115
Objektredundanz	440
OCFS2	241
Okta	130
Open Source	24
Open vSwitch	
<i>installieren</i>	219
<i>Komponenten</i>	220
<i>Konfigurationsbeispiele</i>	220
<i>Schwächen</i>	218
<i>Stärken</i>	218
Open vSwitch (OVS)	218
OpenFabric-Netzwerk	597
<i>prüfen</i>	600
OpenID Connect (OIDC)	125, 129
Open-iSCSI-Initiator	233
<i>konfigurieren</i>	234
OpenSSH	53
OpenVZ	347
OSD (Object Storage Daemon)	443
OSD (Object Storage Device)	443
Overlay-Image	535
OVF/OVA	384

P

PAM (Pluggable Authentication Module)	124
Passthrough	328
Passwörter	289
Paxos	446
pct restore	168
Perl	549
PG Autoscaler	454, 478
phpIPAM	627
Physical Layer	579
Placement Group	453
Placement Group Autoscaler	448
pmxcfs	22, 392, 513
Pool	116, 477
Port-Channel	221
Postfix	50

raw	228	SMART-Werte	176
Realm	122, 123	SMTP	50
Recovery Keys	133	Snapshot	239
Recovery Time Objective (RTO)	505, 506	SNAT	616, 626
Redundanz	508	Snippet	483
Remote-Migration	431	Softdog	512
<i>optimieren</i>	433	Software-Defined Networking (SDN) ...	223, 609
<i>starten</i>	433	<i>automatisierte CI/CD-Pipeline</i>	630
<i>vorbereiten</i>	432	<i>Beispiele</i>	628
Remote-Sync	576	EVPN-Controller	625
Replicated Pool	452	EVPN-Zone	622
replicated_rule (CRUSH)	445	Firewall	143
Replikation	440	FRR	625
Repositories	65, 92	Grundlagen	609
Resilvering	277	<i>in Proxmox VE</i>	610
Resource-Mappings	336	Komponenten	613
Ressourcenaffinität	518	Multi-Tenant-Umgebung	629
REST-API	22	NAT	625
RESTful		Netzwerkanbindung	611
Authentifizierung	541	QinQ-Zone	619
HTTP-Verben	540	Routing	625
Ressource	540	Routing-Controller	623
Statelessness	541	Simple Zone	616
Restore	165, 570	Subnet	616
Retention	158	Testumgebung	629
Retransmit	47	VLAN-Zone	618
Ring-Topologie	591	VNET	615
Roadmap	31	VXLAN-Zone	620
RocksDB	461	Zone	614
Role-Based Access Control (RBAC)	111	Zone-Types	616
Rollback	229	Software-Defined Storage (SDS)	437
Rolle	114	Software-RAID	72
rootfs	352	Source-NAT (SNAT)	614
RPC (Remote Procedure Calls)	286	Spanning-Tree-Protokoll	592
S		Spare-Disk	439
Scrubbing	270, 498	Spectre	312
SCSI-Controller	299	Speicherdeduplizierung	53, 198
Secure Boot	299	Speicherreplikation → Storage-Replikation	
Security Groups	144	SPICE	41
Segmentierung	603	spiceproxy	41
Self-Fencing	515	Split Brain	44, 397
Services	38	ssh	53
Shared Directories	323	Stacking	586
Shared Storage für Cluster	231	Stateless Address Autoconfiguration	
Shutdown Policy	522	(SLAAC)	363
Sicherheit	196	Stern-Topologie	588
smartctl	176	Storage	
RAID-Controller	178	Block-Storage	226
		Durchsatz	251
		File-Storage	225

Storage (Forts.)	
IOPS (Input/Output Operations Per	
Second)	252
Latenz	253
Leistung messen	251
Leistung optimieren	251
lokal für Server	229
Migration	427
Plugins	226
Shared Storage	231
Throughput	251
Typen	226
Storage Area Network (SAN)	232
Storage-Replikation	499
Best Practices	504
einrichten	502
Failover	501
Grundlagen	500
Voraussetzungen	502
Subnet	616
Subscription-Level	373
Subscription-Management	376
Subscriptions	371
auslesen	377
auswählen	374
Cluster-Umgebungen	375
Grundlagen	371
Single-Server-Umgebungen	375
zurücksetzen	377
Suspend Mode	320
Swapping	194
Switch	62
systemd	488
System-Disk	59
systemd-timesyncd	46
Systemvoraussetzungen	66

T

Tape-Backup	576
Task	195
Task (Hintergrundprozess)	188
Task-Log	188
CLI	190
Funktionsweise	188
Web-GUI	189
Template	538
VM	534
Template-Image	535
Templates	350, 533
Thick Provisioning	237

Thin Provisioning	237
Thin-provisioned Volumes	285
Token	112
Topologie	588
TOTP (zeitbasiertes Einmalpasswort)	132
Transaktion	252
Transmit Load Balancing (TLB)	211
TRIM	285
nach Migration	430
Trimming	269
Trunk	60, 221
Trusted Platform Module (TPM)	299, 300
Turnkey-Linux-LXC-Appliances	537

U

Ubuntu	38
Unattended Installation	88
Unique Process ID (UPID)	551
Status überwachen	551
Updates	92
Upgrades	92
User-Mode-iSCSI-Backend	233

V

Vererbung (RBAC)	117
Verify-Jobs	575
Virgil3D	344
VirGL	281, 345
VirGL-GPU	345
VirtIO	281
andere Betriebssysteme	282
Linux	282
Migration	390
Windows	282
virtio-blk	281
VirtIO-FS	323
VirtIO-fs	281
virtiofs-Treiber	326
VirtIO-GPU	281
Virtio-GPU	344
virtio-net	281
virtio-rng	281
virtio-scsi	281
VirtIO-Win	285
Virtualisierung	33
VLAN	214
dedizierte Bridges	604
Guest-Based VLAN-Tagging	605
Layer-3/4-Isolation	606

VLAN (Forts.)	
<i>VM-Interfaces</i>	603
VLAN-Zone	618
VM Creation Wizard	292
VMA-Archive	150
vmdk	228
vmgenid	166
VMware vSphere	33, 382
VNet	615
Volume Shadow Copy Service (VSS)	284
VT-x	55
vttysh	600
VXLAN	614, 620
VXLAN Network Identifier (VNI)	620
vzdump	149, 169, 555
<i>Best Practices</i>	169
<i>Features</i>	149

W

Wake on LAN	193
Wartungsarbeiten	522
Watchdog	511
<i>Softdog</i>	512
WebAuthn	132
Webhooks	50
Wiederherstellung	165
Wiki	28
WinFSP	326
WireGuard	622
Worker-Thread	188, 195
World Wide Name	244

X

X.509-Serverzertifikat	181
x86-64-ABI	309

Y

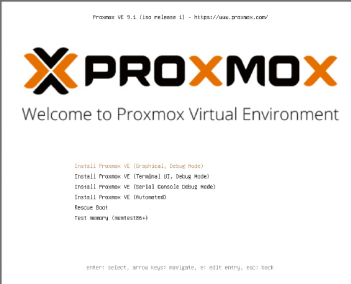
Yubico OTP	133
------------	-----

Z

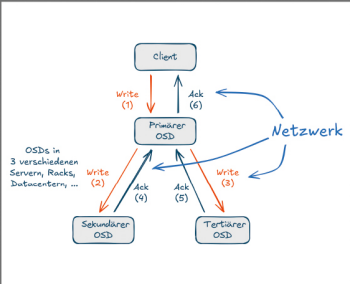
Zeitabgleich	404
Zeitsynchronisation	46
Zero Trust	570
Zertifikate	181
<i>Challenge-Response</i>	183
ZFS	58, 72
<i>defekte Disk tauschen</i>	271
<i>Grundlagen</i>	261
<i>Hardware-Anforderungen</i>	262
<i>Pool importieren</i>	277
<i>Pools</i>	264
<i>Scrubbing</i>	270
<i>Trimming</i>	269
<i>Troubleshooting</i>	271
Zwei-Faktor-Authentifizierung (2FA)	132

Virtualisierte Infrastrukturen administrieren

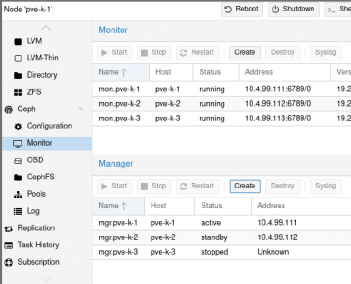
Flexible und leistungsstarke Infrastrukturen sind nicht nur in der Cloud zu finden! Proxmox-Experte Marco Gabriel zeigt Ihnen von der Architekturplanung über den Shared Storage und das SDN, wie Sie die Kontrolle über Ihre IT behalten. So bauen Sie virtualisierte Umgebungen auf, die genau auf Ihre Anforderungen zugeschnitten sind.



Installieren & einrichten



Netzwerke planen



Automation und Kontrolle

Best Practices für Enterprise-Umgebungen

Proxmox macht Sie unabhängig von Cloud-Systemen und spart Lizenzkosten teurer Anbieter. Umso wichtiger ist es, eigenes Know-how aufzubauen: Von den Grundlagen an lernen Sie die Administration Ihrer Proxmox-Infrastruktur.

Von den Grundlagen bis zur Praxis

Verschaffen Sie sich zunächst einen Überblick über die Konzepte. Anschließend geht es um den Betrieb komplexer Cluster: Sie sorgen für Hochverfügbarkeit, richten Backups und Snapshots ein und kümmern sich um das Rechte-Management und die Sicherheit.

Zuverlässig und automatisiert

Ob Sie Storage-Systeme mit Ceph aufbauen oder PBS für Backups nutzen möchten, ob es um Software-Defined Networking oder die Automatisierung geht: Auch fortgeschrittene Themen werden praxisnah behandelt und erklärt.



Marco Gabriel ist der erste offiziell von Proxmox Server Solutions zertifizierte Trainer. Seit mehr als 15 Jahren unterstützt er IT-Fachkräfte bei der Installation und Administration von Proxmox VE und allen individuellen Anforderungen rund um moderne Cluster-Systeme.

Aus dem Inhalt

Architektur und Hardware

Installation, Aktualisierung und Administration

Netzwerkkonfiguration

VMs und Container im Einsatz

Backups mit v2dump und dem Proxmox Backup Server

Cluster-Verwaltung und Lastenmanagment

Shared Storage verwalten

Hyperkonvergenz mit Ceph Storage

Live-Migrationen im und außerhalb des Clusters

Hochverfügbarkeit: Proxmox HA Cluster

Software-Defined Networking (SDN)

