Grundlagen

Virtuelle Maschinen (VM), LXC Container (privilegiert und unprivilegiert) in Proxmox

In diesem Dokument werden die grundlegenden Konzepte von virtuellen Maschinen (VM) sowie privilegierten und unprivilegierten LXC Containern im Kontext der Virtualisierungsplattform [Proxmox](https://www.proxmox.com) erläutert. Zudem werden Vor- und Nachteile dargestellt und Entscheidungshilfen gegeben, wann welche Technologie eingesetzt werden sollte.

Was ist eine Virtuelle Maschine (VM)?

Eine Virtuelle Maschine ist eine vollständig isolierte Umgebung, die ein komplettes Betriebssystem inklusive eigener Kernel-Instanz ausführt. Die Virtualisierung erfolgt in der Regel durch einen Hypervisor (bei Proxmox KVM-basiert). VMs erhalten dedizierte virtuelle Hardware (CPU, RAM, virtuelle Festplatten, virtuelle Netzwerkadapter), auf welcher ein Gastbetriebssystem wie Linux, Windows oder BSD installiert werden kann.

Eigenschaften einer VM:

- Volle Virtualisierung von Hardware-Ressourcen (CPU, Speicher, Netzwerk, Speichergeräte)
- Läuft in einem isolierten Software-Container, der durch den Hypervisor kontrolliert wird
- Betriebssystemunabhängig (kann verschiedenste OS-Typen nutzen)
- Höherer Overhead, da vollständige virtuelle Hardware emuliert wird

Beispiel: Auf einem Proxmox-Host wird eine VM mit 4 vCPUs, 8 GB RAM und 100 GB virtuellem Speicherplatz erstellt, auf der ein vollständiges Ubuntu Server OS läuft.

Was ist ein LXC Container?

LXC (Linux Containers) ist eine leichtgewichtige Virtualisierungs- bzw. Containerisierungstechnologie, die auf Betriebssystem-Ebene arbeitet. Anstatt eine komplette Hardware-Umgebung zu emulieren, nutzen LXC Container den Kernel des Host-Systems direkt, teilen sich also denselben Kernel. Jeder Container ist jedoch in Bezug auf Prozesse, Dateisystem, Netzwerk und Ressourcen weitgehend isoliert.

Eigenschaften von LXC Containern:

- Betriebssystem-Ebene-Isolation (kein eigener Kernel im Container)
- Niedriger Overhead, da keine volle Hardwarevirtualisierung stattfindet
- Sehr schnelle Start- und Stoppzeiten
- In Proxmox direkt unterstützt

LXC Container sind in zwei Varianten verfügbar: **privilegiert** und **unprivilegiert**.

Privilegierte LXC Container

Ein privilegierter Container läuft mit root-Rechten im Kontext des Host-Systems (UID 0 im Container entspricht UID 0 auf dem Host). Dies macht ihn konzeptionell einfacher, birgt aber gewisse Sicherheitsrisiken, da ein Fehler in einem privilegierten Container theoretisch einfacher auf den Host übergreifen kann.

Eigenschaften privilegierter Container:

- Einfachere Verwaltung von Berechtigungen und Ressourcen
- Höheres Sicherheitsrisiko bei Exploits oder Fehlkonfigurationen, da „root“ im Container auch „root“ auf dem Host ist
- Noch immer geringerer Overhead als eine VM

Unprivilegierte LXC Container

Unprivilegierte Container verwenden eine UID/GID-Map (Benutzer- und Gruppen-ID-Zuordnungen) im Host, um sicherzustellen, dass „root“ im Container nicht dasselbe wie „root“ im Host ist. Der Container-„root“ ist hierbei im Host lediglich ein normaler unprivilegierter User, was die Sicherheit deutlich erhöht.

Eigenschaften unprivilegierter Container:

- Höhere Sicherheit, da Host-Root-Rechte nicht direkt auf den Container-Root abgebildet werden
- Etwas aufwändigere Einrichtung, da UID/GID-Maps korrekt konfiguriert sein müssen
- Sicherheit steht hier klar im Vordergrund

Vergleich zwischen VM und LXC

Technologie	Virtualisierungsebene	Kernel-Nutzung	Overhead	Startzeit	Anwendungsfälle
VM (KVM)	Hardware-Ebene	Eigener Kernel im Gast	Höher	Langsamer (Sekunden bis Minuten)	Vollständige OS-Isolation, heterogene Betriebssysteme (z. B. Windows, Linux gemischt)
LXC Container (priv./unpriv.)	Betriebssystem-Ebene	Teilt sich den Kernel mit dem Host	Sehr gering	Sehr schnell (Sekundenbruchteile)	Homogene Linux-Workloads, schnelle Bereitstellung, leichtgewichtige Services

Vor- und Nachteile

Virtuelle Maschinen

Vorteile:

- Volle Isolation, vollständige virtuelle Hardwareumgebung
- Beliebige Betriebssysteme können installiert werden
- Stabiler und etablierter Standard für viele Einsatzzwecke

Nachteile:

- Höherer Ressourcenverbrauch
- Längere Boot-Zeiten
- Komplexere Verwaltung, da eigenes OS gepflegt werden muss

Privilegierte LXC Container

Vorteile:

- Sehr geringe Ressourcenbelastung
- Schnelle Startzeiten
- Einfache Einrichtung im Vergleich zu unprivilegierten Containern

Nachteile:

- Geringere Sicherheit als unprivilegierte Container
- Risiko von Privilege Escalation, falls Lücken ausgenutzt werden

Unprivilegierte LXC Container

Vorteile:

- Höchste Sicherheit unter den Container-Varianten
- Geringe Ressourcenbelastung, schnelle Startzeiten

Nachteile:

- Etwas komplexere Einrichtung durch UID/GID-Mapping
- Teilweise Einschränkungen bei Zugriffen und Berechtigungen, die man im privilegierten Modus nicht hat

Wann was verwenden?

Virtuelle Maschinen: Setze VMs ein, wenn

- Verschiedene Betriebssysteme (z. B. Windows und Linux) auf demselben Host laufen sollen
- Eine vollständige Hardwarevirtualisierung und maximale Isolation erforderlich ist
- Eine stabil etablierte und flexible Lösung benötigt wird, unabhängig vom Host-Kernel

Privilegierte LXC Container: Setze privilegierte Container ein, wenn

- Du schnell und ressourcensparend Linux-basierte Services bereitstellen willst
- Du weniger Aufwand bei der Verwaltung von UID/GID-Maps betreiben möchtest
- Sicherheitsrisiko tolerierbar ist (z. B. interne Entwicklungsumgebung)

Unprivilegierte LXC Container: Setze unprivilegierte Container ein, wenn

- Sicherheit oberste Priorität hat
- Du nur Linux-Workloads betreibst, die im gleichen Kernelumfeld funktionieren
- Du die Vorteile der Containerisierung nutzen willst, ohne die Host-Sicherheit zu kompromittieren

Zusammenfassung

Proxmox bietet mit KVM-VMs und LXC-Containern zwei Hauptvirtualisierungsansätze. VMs sind ideal für heterogene, vollständig isolierte Umgebungen und bieten maximale Flexibilität, aber auf Kosten von Ressourcen und Startzeiten. LXC Container bieten leichtgewichtigere, schnellere Umgebungen für Linux-basierte Workloads. Unprivilegierte Container erhöhen dabei die Sicherheit erheblich.

Durch die gezielte Wahl der passenden Technologie kannst Du Deine Ressourcen effizient nutzen und Deine Sicherheits- und Performanzanforderungen optimal ausbalancieren.

From:
https://wiki.mahlen.biz/ - **Smart-Home Wiki**

Permanent link:
https://wiki.mahlen.biz/doku.php?id=proxmox:proxmox_grundlagen

Last update: **10.12.2024**