OpenVZ

Servervirtualisierung Open Source Project

Einführung

OpenVZ ist eine auf Linux aufbauende Servervirtualisierungslösung auf Betriebssystemebene. Sie erstellt mehrere isolierte, sichere Virtual Environments (VEs, auch bekannt als Virtuelle Private Servers oder VPS) auf einem einzigen physikalischen Server, sorgt so für eine bessere Serverausnutzung und stellt sicher, dass Applikationen nicht miteinander in Konflikt geraten. Jedes VE arbeitet und verhält sich genau wie ein eigenständiger Server; VEs können unabhängig voneinander neu gestartet werden und haben Root-Zugriff, Benutzer, IP-Adressen, Speicher, Prozesse, Dateien, Applikationen, Systembibliotheken und Konfigurationsdateien.

Mit OpenVZ können Sie mehrere hundert VEs auf einem einzigen physikalischen Rechner erstellen (siehe Diagramm rechts).

Technischer Hintergrund

OpenVZ besteht aus einem modifizierten Linux-Kernel, erweitert um die Funktionalitäten zur Erstellung von VEs, sowie Tools auf Benutzerebene. Der OpenVZ Kernel ermöglicht Virtualisierung, Isolierung und Ressourcenverwaltung.

Virtualisierung und Isolierung

Jedes VE verfügt über:
- eigene Dateien (Systembibliotheken, Applikationen, virtualisierte /proc und /sys, Dateisperren)
- eigene Prozess-Struktur (mit virtualisierten PIDs)
- eigenes Netzwerk (virtualisiertes Netzwerk-Device, IP-Adressen, Routing- und Netzfilter-(iptables)Regeln)
- eigene Geräte (falls nötig, kann einem VE Zugriff auf reale Geräte wie Netzwerkschnittstellen, Festplattenpartitionen, serielle Schnittstellen etc. gewährt werden)
- IPC(inter-process communication)-Objekte (gemeinsam genutzter Speicher, Semaphore, Nachrichten)

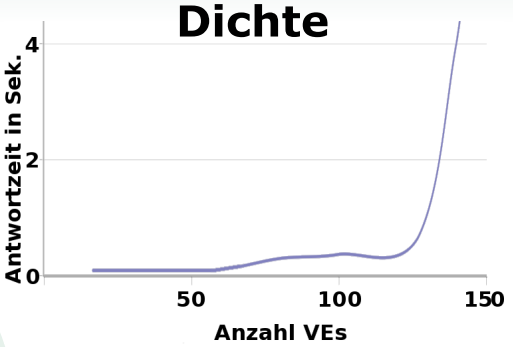
Ressourcenverwaltung

Der Kernel verteilt und begrenzt VE-Ressourcen, so dass kein VE Systemressourcen missbrauchen kann. Die drei Haupt-Subsysteme sind:
- User Beancounters. Ein Set von pro-VE-Ressourcenzählern, Limits und Garantien. Ressourcen wie Kernelspeicher, Netzwerkpuffer, physikalische und virtuelle Speicherseiten etc. werden berechnet. Man könnte also von einem erweiterten "ulimit" sprechen.
- CPU-Planer. Er balanciert die CPU-Zeit zwischen VEs aus, damit kein VE die CPU missbrauchen kann, und kann verwendet werden, um CPU-Limits und Garantien zu gewähren.
- Zwei-Ebenen-Festplattenquota. Die erste Ebene ist eine pro-VE-Festplattenquota, die zweite Ebene eine standardmäßige UNIX Festplattenquota pro Benutzer und pro Gruppe innerhalb eines VE.

Live-Migration und Checkpointing

Der Kernel wird eingefroren und speichert den kompletten Zustand (Checkpoint) eines VE auf einem physikalischen Server, verschiebt ihn auf einen anderen und stellt ihn dort wieder her und behält dabei alles einschließlich offener Netzwerkverbindungen bei. Aus Sicht des Benutzers sieht es wie eine leichte Verzögerung im Prozessablauf, nicht jedoch wie eine Downtime aus.

Dichte



Das obige Diagramm zeigt die Abhängigkeit der Antwortzeit von der Anzahl der laufenden VEs an. Gemessen wurde die Antwortzeit von Apache-Webservern auf einem Rechner mit 768 MB RAM. Außerdem lief auf jedem VE init, syslogd, sendmail, sshd und cron. Eine Antwortzeit unter einer Sekunde wird als normal angesehen. Bei mehr als 120 VEs wird die Antwortzeit wegen enormer Auslagerung inakzeptabel. Auf dem gleichen Rechner mit 2 GB RAM können schätzungsweise 320 derartige VEs laufen.

Einsatzbeispiele

Serverkonsolidierung

- Einheitliche Verwaltung
- Leichtes Upgrade
- Skalierbarer
- Schnelle Migration
- Einsparung von Strom und Rackspace

Entwicklung und Test

- Parallelbetrieb mehrerer Distributionen
- Ein VE kann innerhalb einer Minute erstellt werden
- Hunderte VEs möglich
- Klonen, Schnappschüsse, Rollbacks
- Ein VE ist eine Sandbox: Arbeiten/Spielen, keine Sicherheitsbedenken

Security

- Jeder Applikation ihr eigenes isoliertes VE
- Sicherheitslücke in einer Applikation beeinflusst keine anderen VEs
- Dynamische Ressourcenverwaltung

Hosting

- Isolierte Benutzer
- Ein VE ist wie ein realer Server, nur billiger
- Leichter zu administrieren

Bildungseinrichtungen

- Jeder Schüler hat Root-Zugriff
- Unterschiedliche Distributionen
- Wenig Hardware benötigt

http://openvz.org/