OpenVZ on CentOS

Frederik Kaputa
Overview

- Background
- Virtualization types on x86
- Introduction to OpenVZ
- OpenVZ Installation / Configuration
- VE Load Demonstration
Virtualization

- Why virtualization
- Technology
- Performance
- Usability
Virtualization Types

Different types of virtualization

- Emulation
- Paravirtualization
- Native Virtualization
- OS Level Virtualization
Emulation

- Emulates the systems hardware
- 100% Emulators can emulate other CPUs
- Emulators use JIT to speed up
- Examples:
  - VMWare, Qemu, Bochs
Emulation

ADVANTAGES

● Can emulate different CPUs
● Unmodified OS as guest

DISADVANTAGES

● Slow(er)
● Limited virtual servers per system
Paravirtualization

- Uses a Hypervisor / Virtual machine monitor
- Guests need to be modified
- Major performance increase
- Examples:
  - Xen, UML
Paravirtualization

ADVANTAGES

• Runs very fast
• Less resource overhead then emulation

DISADVANTAGES

• Modified Guest kernel / drivers
• ( Limited Guests )
Native Virtualization

- Intel VT, AMD-V
- Guests no longer need to be modified
- A bit slower than Paravirtualization
- Examples:
  - Xen, KVM
OS Level Virtualization

- Also known as containers
- Share the same kernel
- Lower overhead
- Best possible performance regarding resource management
- Examples:
  - Solaris Zones
  - Linux-VServer
  - OpenVZ
OS Level Virtualization

ADVANTAGES

• Highest density of running guest
• Native speed

DISADVANTAGES

• Runs 1 kernel for all VE/VPS
• Less suitable for testing purposes
Comparison

**Emulation**
- Emulates HW, Test platform, Slowest

**Para/native virtualization**
- Higher performance, Limited # servers

**OS Level**
- High density, Native speed, Resource management, Single kernel
OpenVZ on CentOS

OpenVZ
OpenVZ

• Runs a modified Linux Kernel
• Works with Virtual Environments (VE)
• Provides utilities for VE / resource management
OpenVZ - VE

• Virtual Environment provides an isolated execution environment

• Looks and feels like a separate physical server

• Has its own processes (init), filesystem, users, network interfaces, routing tables, firewall rules
Modified Linux kernel with the following additional features:

- Enables virtualization and isolation of VE
- Resource management (subsystem limits)
  - Two level disk quota
  - “Fair” CPU scheduler
  - User Beancounters
- Checkpointing (freezing)
OpenVZ - Installation

- Kernel installation via YUM / RPM
- VZ Tools installation
- Template installation

Quick install guide:
http://wiki.openvz.org/Quick_installation
OpenVZ - Templates

• Building Block for VEs
• Template metadata
  • List of packages included
  • Location of package repositories
  • Distribution specific scripts
• Template Cache
  • Precreated template usually from the template metadata
  • Delivers faster provisioning of new VE
Use Cases

- Server consolidation
- Development and testing
- Educational
- Hosting
Use Cases

- Numproc, numtcpsock
- numproc
- Educational
- Hosting