

PROXMOX VE

PRIMER ON A VMWARE ALTERNATIVE



INTRODUCTION

WHAT IS PROXMOX?

- Proxmox Virtual Environment (Proxmox VE) is an open-source virtualization platform that integrates multiple open-source technologies.
- Initial release, 2008.

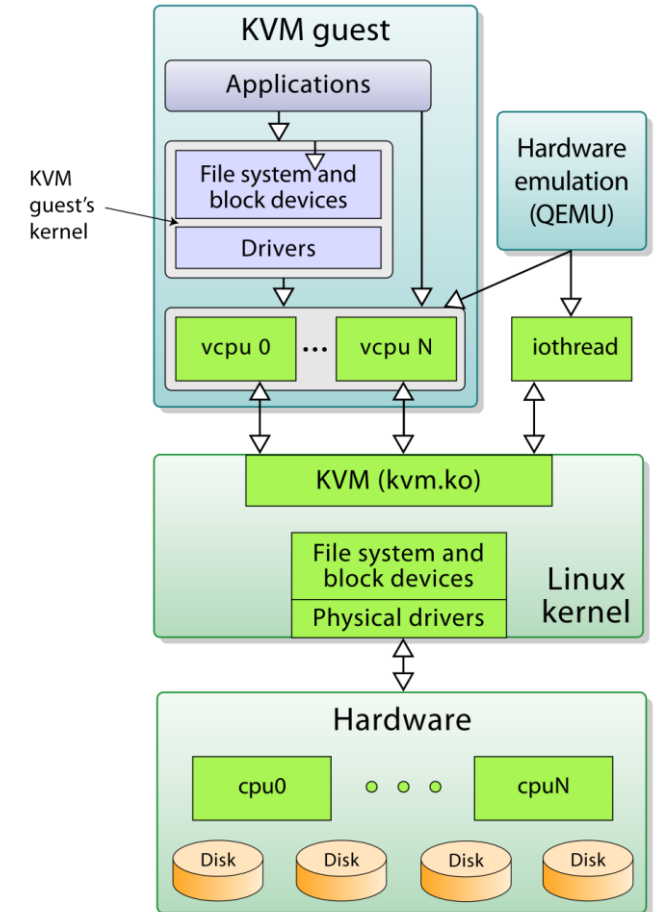
HYPERVERSOR TECHNOLOGY

QEMU/KVM + vFIO

Comparable
VMware Product
-ESXi



Type 1 Hypervisor

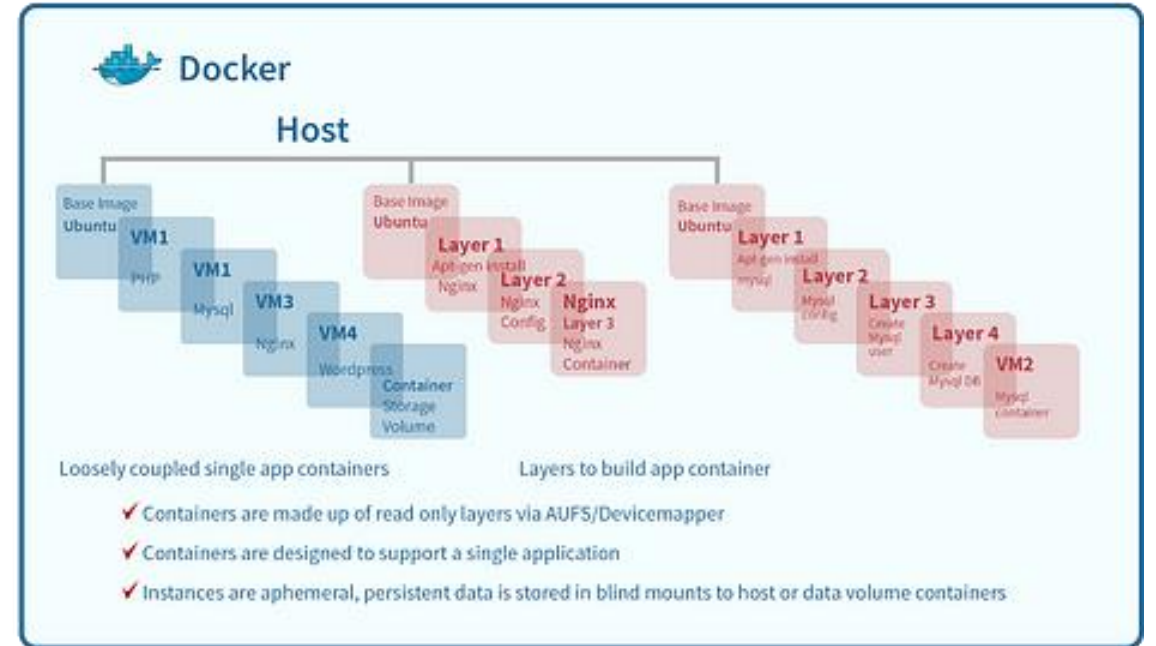


CONTAINER TECHNOLOGY

LXC

Key differences between LXC and Docker

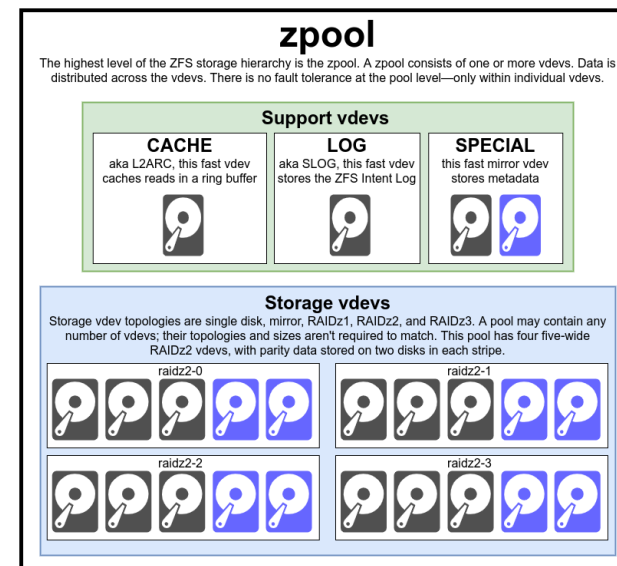
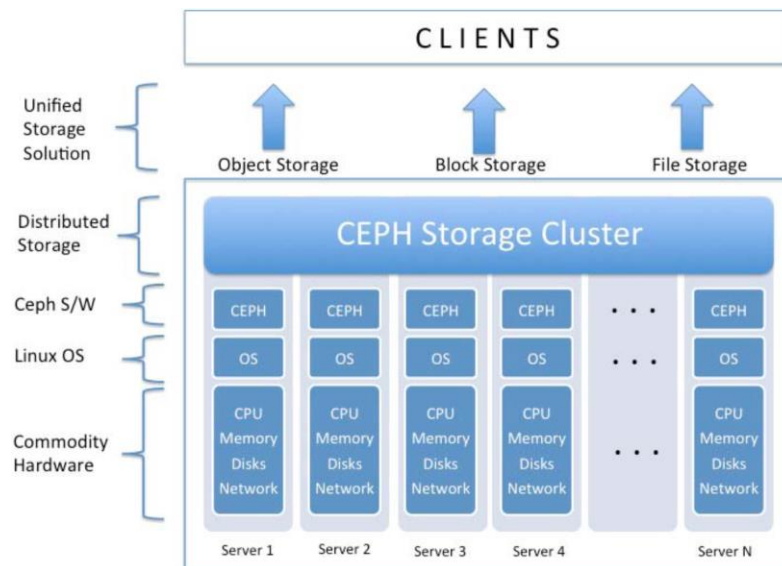
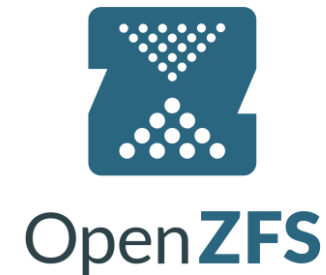
Comparable
VMware Product
- vSphere Integrated
Container
(Replaced by Tanzu)



STORAGE

CEPH + ZFS + BLOCK

Comparable
VMware Product
- Datastores
- vSAN

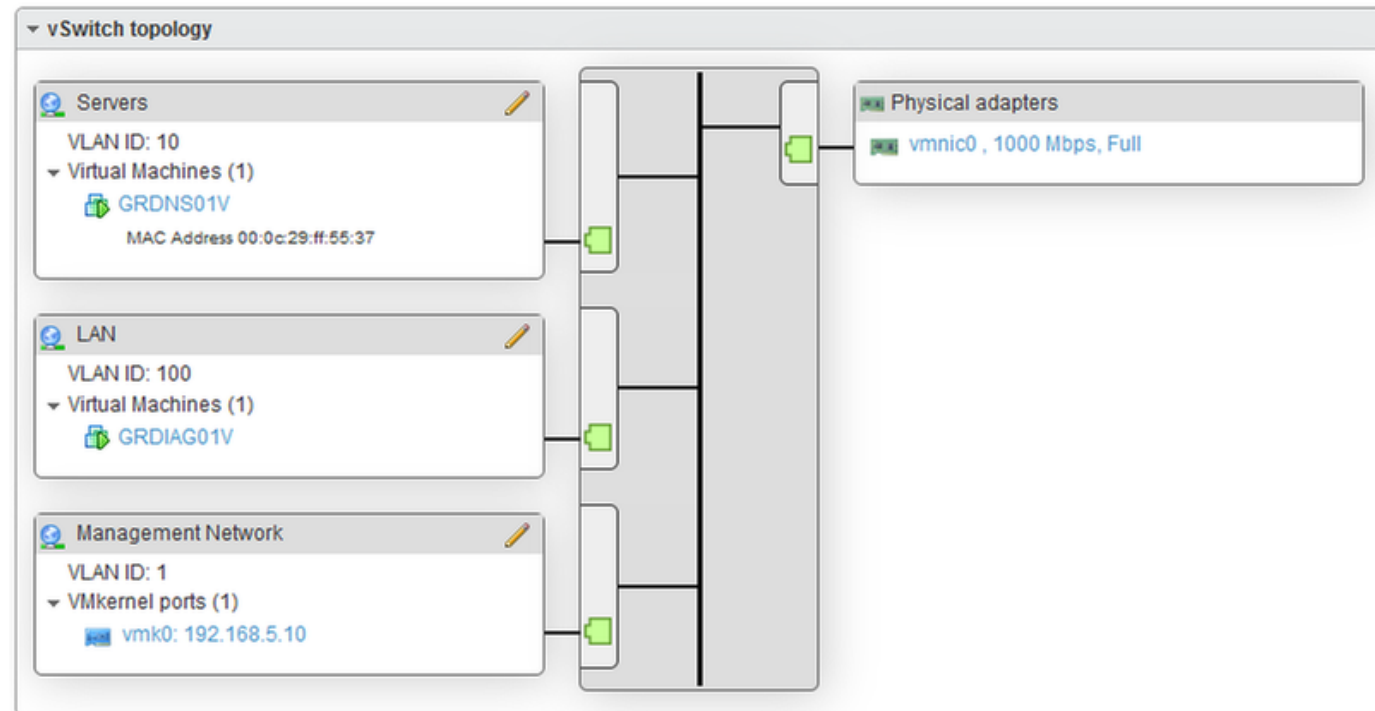


NETWORKING

OPEN VSWITCH

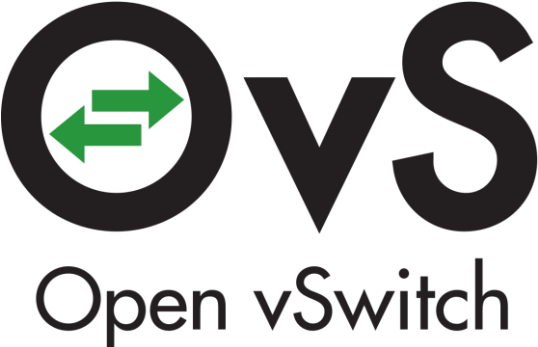
Comparable
VMware Product

- vSphere HA
- vSwitch
- NSX



NETWORKING

OPEN VSWITCH



The image displays three screenshots of the OVS TOOLBOX application interface, illustrating different views of the Open vSwitch configuration.

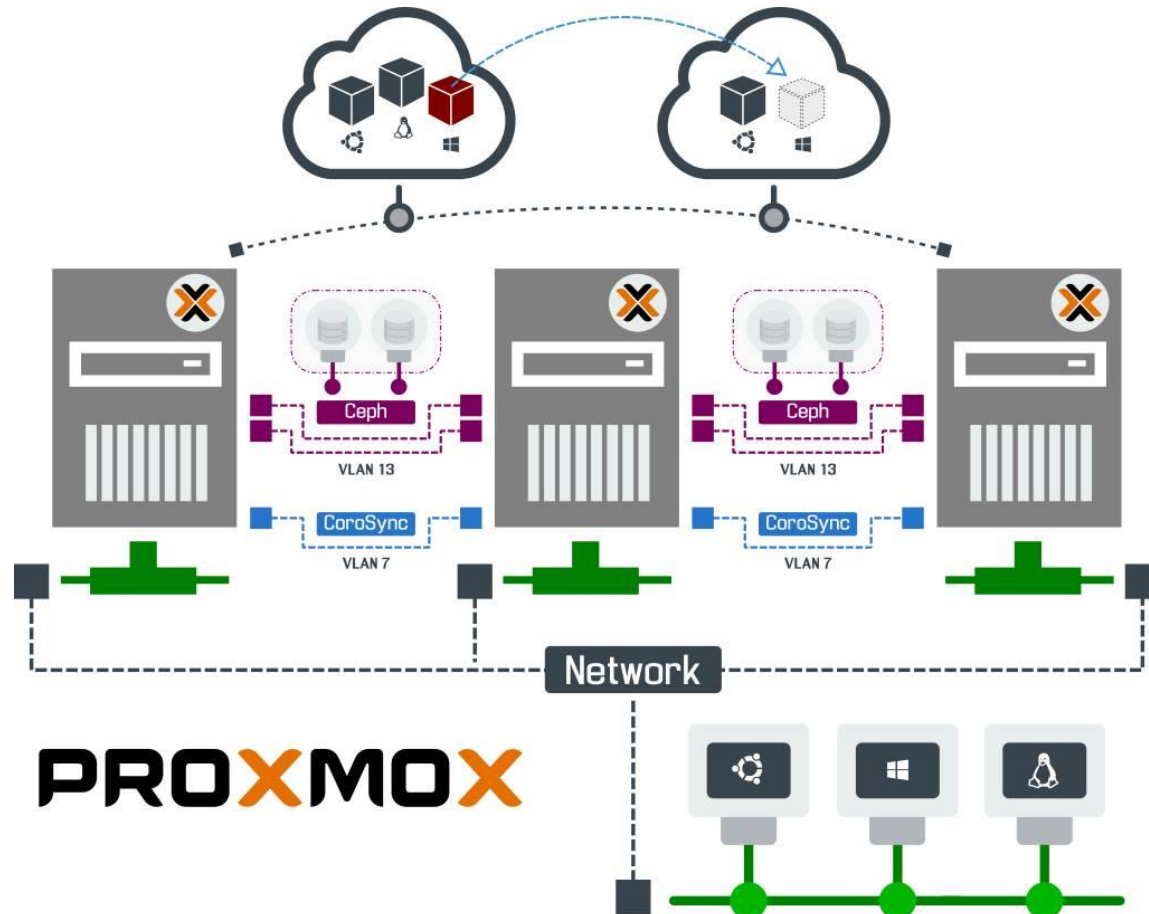
- Left Screenshot:** Shows the 'queues' tab. A table lists queue configurations with columns for 'queue uuid', 'min-rate', 'max-rate', 'burst', 'priority', and 'dscp'.

queue uuid	min-rate	max-rate	burst	priority	dscp
415f5dd8-d979-4330-8a10-bbdaa8ed956d		20000000			
29190a83-cc28-4e61-b4d2-66aaee7475a9		10000000			
- Middle Screenshot:** Shows the 'Port/interface' configuration for port 'P01'. It displays various parameters such as 'link_speed', 'duplex', 'MTU', and 'admin state'. The 'Status and statistics' section shows details for the port's address and current state.
- Right Screenshot:** Shows a network diagram with a central bridge 'br2 (openvswitch)'. It is connected to several virtual machines (VMs) via 'eth0 (veth)' interfaces. The VMs are labeled with their PID and IP addresses, and are connected to the bridge through 'tag 100'.

HA / SCALABILITY

Corosync + Pacemaker

Comparable
VMware Product
- vCenter



MANAGEMENT

PROXMOX VE WEB GUI

Comparable
VMware Product

- vCenter
- vSphere

The screenshot displays the Proxmox VE Web GUI interface. The top navigation bar includes 'Documentation', 'Create VM', 'Create CT', and the user 'root@pam'. The main content area is titled 'Node 'pve3'' and shows the Ceph status. The Ceph Version is 17.2.5. The Status section features a large green progress indicator and a table for OSDs:

	In	Out
Up	4	0
Down	0	1
Total:	5	

PGs: 33, active+clean: 33. The Services section shows Monitors (pve1, pve2, pve3), Managers (pve1, pve2, pve3), and Meta Data Servers, all with green checkmarks. The Performance section is partially visible. At the bottom, a 'Tasks' table shows recent console sessions:

Start Time	End Time	Node	User name	Description	Status
Feb 06 18:09:16	Feb 06 18:58:04	pve1	root@pam	VM/CT 102 - Console	OK
Feb 06 15:14:56	Feb 06 15:14:56	pve1	root@pam	VM/CT 102 - Console	OK
Feb 06 15:13:47	Feb 06 15:13:48	pve1	root@pam	VM/CT 102 - Console	OK
Feb 06 15:13:32	Feb 06 15:13:33	pve1	root@pam	VM/CT 102 - Console	OK
Feb 06 15:12:59	Feb 06 15:13:02	pve1	root@pam	VM/CT 102 - Console	OK

AUTOMATION

CLOUD-INIT + API



cloud-init

PROXMOX Virtual Environment 7.2-4

Server View

Virtual Machine 382204 (ubuntu-cloud-jammy) on node 'pmvm3'

Summary

Hardware

Cloud-Init

Options

Task History

VM 382204

Migrate

Clone

Firewall

Permissions

Remove Edit Regenerate Image

User	sysadmin
Password	*****
DNS domain	lab.netcave.io
DNS servers	use host settings
SSH public key	user@gram17.bash userg@gram17.powershell robert@mini-ryzen9-powershell robert@mini-ryzen9-bash
IP Config (net0)	ip=dhcp

AUTOMATION

CLOUD-INIT + API

The screenshot displays a REST API client interface. On the left is a 'Resource Tree' with a search bar and a list of folders including 'access', 'cluster', 'nodes', and various sub-folders like 'apt', 'capabilities', 'ceph', etc. The main panel shows the details for the endpoint `POST /nodes/{node}/hosts`. It includes a 'Description' section with the text 'Write /etc/hosts.', a 'Usage' section with HTTP and CLI examples, and a 'Parameters' table. The parameters table is divided into 'Required' and 'Optional' sections. The 'Required' section lists 'data' and 'node', both of type 'string'. The 'Optional' section lists 'digest', also of type 'string'. Below the parameters is a 'Returns: null' section and a 'Required permissions' section with a 'Check' field containing the JSON array `["perm", "/nodes/{node}", ["Sys.Modify"]]`.

Resource Tree

Path: `/nodes/{node}/hosts`

GET POST

Description

Write `/etc/hosts`.

Usage

HTTP: `POST /api2/json/nodes/{node}/hosts`

CLI: `pvesh create /nodes/{node}/hosts`

Parameters

Name ↑	Type	Default	Format	Description
Required				
<code>data</code>	string		<code><string></code>	The target content of <code>/etc/hosts</code> .
<code>node</code>	string		<code><string></code>	The cluster node name.
Optional				
<code>digest</code>	string		<code><string></code>	Prevent changes if current configuration file has a different digest. This can be used to prevent concurrent modifications.

Returns: null

Name ↑	Type	Default	Format	Description
--------	------	---------	--------	-------------

Show RAW

Required permissions

Check: `["perm", "/nodes/{node}", ["Sys.Modify"]]`

AUTOMATION

ANSIBLE+ TERRAFORM



THANK YOU