

Features

14-18 minutes

Proxmox Virtual Environment is a powerful open-source server virtualization platform to manage two virtualization technologies - KVM (Kernel-based Virtual Machine) for virtual machines and LXC for containers - with a single web-based interface. It also integrates out-of-the-box-tools for configuring high availability between servers, software-defined storage, networking, and disaster recovery.

Highlights

- [Virtualization](#)
- [Management](#)
- [Clustering](#)
- [Authentication](#)
- [HA](#)
- [Network](#)
- [Storage](#)
- [Ceph](#)
- [Firewall](#)
- [Backup](#)
- [Backup Server](#)

Server Virtualization

Proxmox Virtual Environment is based on Debian GNU/Linux and uses a custom Linux Kernel. The Proxmox VE source code is free, released under the [GNU Affero General Public License, v3](#) (GNU AGPLv3). This means that you are free to use the software, inspect the source code at any time and contribute to the project yourself. You can download the Proxmox VE ISO installer or inspect the code in the public [code repository \(git\)](#).

Using open-source software guarantees full access to all functionality, as well as a high level of reliability and security. We encourage everybody to contribute to the Proxmox VE project, while Proxmox, the company behind it, ensures that the product meets consistent, enterprise-class quality criteria.

[Read more about the Proxmox technology](#)

Kernel-based Virtual Machine (KVM)

KVM is the industry-leading Linux virtualization technology for full virtualization. It's a kernel module, that's merged into the mainline Linux kernel, and it runs with near-native performance on all x86 hardware with virtualization support—either Intel VT-x or AMD-V.

With KVM you can run both Windows and Linux in virtual machines (VMs), where each VM has private, virtualized hardware: a network card, disk, graphics adapter, etc. Running several applications in VMs on a single system, enables you to save power and reduce costs, while at the same time, giving you the

flexibility to build an agile and scalable software-defined data center, that meets your business demands.

Proxmox VE has included KVM support since the beginning of the project, back in 2008 (that is since version 0.9beta2).

[Read more about KVM](#)

Container-based virtualization

Container-based virtualization technology is a lightweight alternative to full machine virtualization, because it shares the host system's kernel.

Linux Containers (LXC)

LXC is an operating-system-level virtualization environment for running multiple, isolated Linux systems on a single Linux control host. LXC works as a userspace interface for the Linux kernel containment features. Users can easily create and manage system or application containers with a powerful API and simple tools.

[Read more about Linux Containers \(LXC\)](#)

Central Management

To manage all tasks of your virtual data center, you can use the central, web-based management interface. The whole functionality of the web interface is also accessible via CLI or REST API, which can be used to automate tasks.

Web-based management interface

Proxmox VE is easy to use. You can do all management tasks with the integrated graphical user interface (GUI), there is no need to install a separate management tool. The central web interface is based on the ExtJS JavaScript framework and can be accessed from any modern browser. In addition to management tasks, it also provides an overview of the task history and system logs of each node. This includes running backup tasks, live migration, software-defined storage, or HA triggered activities. The multi-master tool allows you to manage your whole cluster from any node of your cluster; you don't need a dedicated manager node.

Proxmox VE mobile

You can [access Proxmox VE on mobile devices](#) either via an Android app or via the HTML5-based mobile version of the web interface. The Proxmox VE Android app is based on the Flutter framework, and allows you to access your Proxmox VE server and manage your cluster, nodes, VMs, and containers. The Proxmox VE HTML5 mobile client enables you to manage Proxmox VE on the go, including access to the SPICE and HTML5 console. This allows you to manage VMs and containers, and view their configuration.

Command line interface (CLI)

For advanced users who are used to the comfort of the Unix shell or Windows Powershell, Proxmox VE provides a command line interface to manage all the components of your virtual environment. This command line interface has intelligent tab completion and full documentation in the form of UNIX man

pages.

REST API

Proxmox VE uses a RESTful API. We chose JSON as the primary data format, and the whole API is formally defined using JSON Schema. This enables fast and easy integration for third party management tools, such as custom hosting environments.

Clustering

While many people start with a single node, Proxmox Virtual Environment can scale out to a large set of clustered nodes. The cluster stack is fully integrated and ships with the default installation.

Proxmox Cluster File System (pmxcfs)

Proxmox VE uses the [unique Proxmox Cluster File System \(pmxcfs\)](#), a database-driven file system developed by Proxmox.

The pmxcfs enables you to synchronize configuration files across your cluster. By using Corosync, these files are replicated in real time to all cluster nodes. The file system stores all data inside a persistent database on disk, nonetheless, a copy of the data resides in RAM. The maximum storage size is currently 30 MB - more than enough to store the configuration of several thousands of VMs.

Proxmox VE is the only virtualization platform using this unique cluster file system, pmxcfs.

Live/Online migration

With the integrated [live/online migration feature](#), you can move running virtual machines from one Proxmox VE cluster node to another, without any downtime or noticeable effect from the end-user side.

Administrators can initiate this process from either the web interface or the command line. This enables you to minimize downtime, in case you need to take the host system offline for maintenance.

Unique multi-master design

To simplify the management of a cluster, you can carry out maintenance tasks cluster-wide, from any node. The integrated web-based management interface gives you a clean overview of all your KVM guests and Linux containers across your cluster. You can easily manage your VMs and containers, storage or cluster from the GUI. There is no need to install a separate, complex, and pricey management server.

Authentication

Role-based administration

You can define granular access to all objects (like VMs, storage, nodes, etc.) by using the role-based permission management system. This allows you to define privileges and helps you to control access to objects. This concept is also known as access control lists: Each permission specifies a subject (a user group, or API token) and a role (set of privileges) on a specific path.

Authentication realms

Proxmox VE supports multiple authentication sources, for example Linux PAM, an integrated Proxmox VE authentication server, LDAP, Microsoft Active Directory, and OpenID Connect.

Proxmox VE High Availability (HA) Cluster

A multi-node Proxmox VE Cluster enables the creation of highly available virtual servers. The Proxmox VE HA Cluster is based on proven Linux HA technologies, providing stable and reliable HA service.

The entire Proxmox VE HA Cluster can be easily configured from the integrated web-based user interface.

Proxmox VE HA Manager

The resource manager, Proxmox VE HA Manager, monitors all VMs and containers in the cluster and automatically comes into action if one of them fails. The Proxmox VE HA Manager works out-of-the-box. Zero configuration is needed. Additionally, the watchdog-based fencing dramatically simplifies deployment.

Proxmox VE HA Simulator

Proxmox VE includes a HA Simulator. This allows you to test the behavior of a real-world 3 node cluster with 6 VMs. The Proxmox HA Simulator runs out-of-the-box and helps you to learn and understand how Proxmox VE HA works.

Read more about the [Proxmox VE High Availability](#)

Networking

Proxmox VE's networking capabilities enable responsive and adaptable networks that can scale according to your business needs.

Software-Defined Network (SDN)

The Software-defined Networking (SDN) feature allows Proxmox VE to manage advanced networking configurations and multitenancy setups across Proxmox VE clusters. Possible use cases range from an isolated private network on each individual node to complex overlay networks across multiple Proxmox VE clusters on different locations.

It is applicable to all sizes of networks, from a simple routed NAT setup, traditional separation into 802.1q VLANs, to features like QinQ, VXLAN tunneling, and BGP-based EVPN infrastructures.

Read more about [Software-Defined Network \(SDN\)](#)

Linux Networking Stack

For simple setups, Proxmox VE provides flexible network configuration options for local nodes. A bridged network model is used to provide seamless connectivity between virtualized environments and the external network. Bridges are comparable to physical network switches, but implemented as software on the Proxmox VE host.

For additional flexibility, you can set up VLANs, create a bond interface, and manage basic network routing.

Read more on the [Proxmox VE Network Configuration](#)

Flexible Storage Options

The Proxmox VE storage model is very flexible. VM images can either be stored on one or several local storage devices or on shared storage like NFS and SAN. There are no limits. You can configure as many storages as you like, and can use all storage technologies available for Debian GNU/Linux. The benefit of storing VMs on shared storage is the ability to live-migrate running machines without any downtime.

In the Proxmox VE web interface, you can add the following storage types:

Network storage types

- LVM Group (network backing with iSCSI targets)
- iSCSI target
- NFS Share
- SMB/CIFS
- Ceph RBD
- Direct to iSCSI LUN
- GlusterFS
- CephFS

Local storage types

- LVM Group
- Directory (storage on an existing filesystem)
- [ZFS](#)

Read more on the [Proxmox VE Storage Model](#)

Software-Defined Storage with Ceph

Ceph is an open-source distributed object store and file system designed to provide excellent performance, reliability and scalability. Proxmox Virtual Environment fully integrates Ceph, giving you the ability to run and manage Ceph storage directly from any of your cluster nodes.

Ceph provides two types of storage, RADOS Block Device (RBD) and CephFS. An RBD provides block level storage, for content such as disk images and snapshots. CephFS implements a POSIX-compliant filesystem using a Ceph storage cluster to store its data.

Proxmox VE Firewall

The built-in Proxmox VE Firewall provides an easy way to protect your IT infrastructure. The firewall is completely customizable, allowing complex configurations via the GUI or CLI.

You can set up firewall rules for all hosts inside a cluster, or define rules for virtual machines and containers only. Features like firewall macros, security groups, IP sets and aliases help to make that task

easier.

Distributed firewall

While all configuration is stored in the cluster file system, the iptables-based firewall runs on each cluster node, and thus provides full isolation between virtual machines. The distributed nature of this system also provides much higher bandwidth than a centralized firewall solution.

IPv4 and IPv6

The firewall has full support for IPv4 and IPv6. IPv6 support is fully transparent, and we filter traffic for both protocols by default. Thus, there is no need to maintain a different set of rules for IPv6.

Read more about the [Proxmox VE Firewall](#)

Backup/Restore

Backups are a basic requirement for any sensible IT environment. The Proxmox VE platform provides a fully integrated solution, using the capabilities of each storage and each guest system type. Backups can be easily started with the GUI or with the vzedump backup tool (via command line). These backups are always full backups - containing the configuration of VMs and container, and all data.

The integrated backup tool (vzedump) creates consistent snapshots of running containers and KVM guests. It basically creates an archive of the VM or container data and also includes the configuration files.

Scheduled backup

Backup jobs can be scheduled so that they are executed automatically on specific days and times, for selectable nodes and guest systems.

Backup storage

KVM live backup works for all storage types including VM images on NFS, iSCSI LUN, and Ceph RBD. The Proxmox VE backup format is optimized for storing VM backups quick and effectively (accounting for sparse files, out of order data, minimized I/O).

Read how to configure [Proxmox VE Backup and Restore](#)

Proxmox Backup Server Integration

[Proxmox Backup Server](#) is our enterprise-class backup solution, that is capable of backing up VMs, containers, and physical hosts. Support for this is fully integrated into Proxmox VE, meaning you can seamlessly back up and restore guests using the same common interface that the other storage types use.

These backups are incremental, only transferring newly changed data over the network. This is highly beneficial in terms of network bandwidth and backup job run time. Data can also be easily encrypted on the client side, so that your backed up data is inaccessible to attackers.

Live-restore

Restoring large backups can take a long time and be a major source of downtime in case of disaster. However, for VM backups that are stored on a Proxmox Backup Server, the live-restore feature minimizes this downtime, allowing the VM to start as soon as the restore begins. In this case, the data is continuously copied in the background, prioritizing chunks that the VM is actively accessing.

Single-file restore

Oftentimes, only a single file or directory is needed from a backup. From the Proxmox VE web interface, you can securely search for and restore individual files or directories from a VM or container backup.

Related Resources

Datasheet

To view the comprehensive feature-set, download the datasheet for Proxmox Virtual Environment.

[Download datasheet](#)

Source code

Proxmox solutions use the free, copyleft license GNU AGPLv3. The source code is available via Git.

[See developer documentation](#)