ProxLB: Intelligent Workload Balancing for Proxmox Clusters

ProxLB is an advanced and DRS alike open-source load balancing solution for Proxmox clusters that addresses the absence of a Dynamic Resource Scheduler. It intelligently distributes workloads across available nodes based on memory, CPU, or disk usage to optimize resource utilization and prevent overprovisioning.

As a fully open-source and free solution, ProxLB provides flexibility through configuration files and API calls, supports maintenance mode for seamless node updates, and implements affinity and antiaffinity rules for optimized VM placement.



Project

- Developed in 2024 by Florian Paul Azim Hoberg (@gyptazy)
- Inspired by the need for such a solution in the BoxyBSD.com project.
 - To balance clusters.
 - To place VMs on nodes with the most available resources.
- Driven by customer demand for a VMware to Proxmox migration solution.
 - Recognizing the absence of such a solution.
- Released ProxLB as an open-source project under the GPL v3 license.









Key Features and Capabilities



Intelligent Workload Distribution

Rebalance VMs and containers across your cluster based on memory, CPU, and local disk usage to optimize resource utilization.



Maintenance Mode

Automatically migrate all workloads from nodes marked for maintenance to other nodes in the cluster, ensuring zero downtime during updates.



Affinity Rules

Define which VMs should run together on the same node or be separated across different nodes for optimal performance and availability.

Secure API Integration

Fully integrated with Proxmox API and ACL system without requiring SSH access, enhancing security and simplifying configuration.



How ProxLB Works

Q ::: \blacksquare

Resource Metrics Collection

Gathers detailed resource usage metrics from all nodes in the cluster through the Proxmox API, including metrics for each VM and container.

Balanciness Evaluation

Evaluates the difference between maximum and minimum resource usage across nodes, initiating rebalancing if this difference exceeds a predefined threshold.

Balancing Matrix Creation

Creates a matrix that sorts VMs by their resource usage based on the selected balancing mode (CPU, memory, or disk).

Recursive Optimization

Places VMs with highest resource usage on nodes with most free resources, running recursively until the operator-defined balanciness is achieved.

Running

O Silceded

Design restorm VM 105 - Stan VM 104 - Done VM 103 - Scient VM 103 - Migrah VM 100 - Start



Installation Options

Debian Package

The simplest installation method for Debian-based systems:

- Add the repository to your sources
- Import the GPG key for verification 2.
- Install the package with apt-get 3.
- Configure and start the service 4.

Container / Docker

Run ProxLB as a containerized application:

- Pull the official image from 1. cr.gyptazy.com
- Download and customize the 2. configuration file
- Mount the config and run the 3. container

From Source

For maximum flexibility or non-Debian systems:

- Clone the GitHub repository
- Install Python dependencies 2.
- Configure and run directly with Python 3.

Installation Methods

Debian Package



Docker Container Installation

Source Code











Source

(Laueres (courser: rack; cerser'; developer

sourtède: developer



ricilis carcorrion is eliferen en come

tester is (repotentierd))

miter (eH): laclouer(class) selettein (latert II)

RANK GARAGO 1-115

Configuration Options



{i}}

ðjð

 \bigcirc

Proxmox API Settings

Configure connection details including hosts, authentication credentials, SSL verification, and timeout settings to securely connect to your Proxmox cluster.

Cluster Configuration

Define maintenance nodes, nodes to ignore during balancing, and whether to allow overprovisioning when distributing workloads across the cluster.

Balancing Parameters

Set balancing behavior including affinity enforcement, migration method (live or shutdown), balanciness threshold, and whether to include VMs with local disks.

Service Settings

Configure daemon mode, scheduling frequency, and logging level to determine how and when ProxLB performs balancing operations.



🗯 Made with Gamma

Affinity & Anti-Affinity Rules

Affinity Rules

Group specific VMs together on the same host using tags with the prefix **plb_affinity_\$TAG**. This is ideal for workloads requiring low-latency communication, such as clustered databases or application servers that frequently exchange data.

Description	Node
118 (dev-talos01) plb_affinity_talos	virt01
121 (dev-talos02) plb_affinity_talos	virt01
123 (dev-talos03) plb_affinity_talos	virt01

Anti-Affinity Rules

Ensure designated VMs run on different physical hosts using tags with the prefix **plb_anti_affinity_\$TAG**. This is particularly useful for high-availability setups where redundancy is crucial to reduce the risk of a single point of failure.

Description	
108 (ntp01.gyptazy.de)	plb_anti_affinity_ntp
109 (ntp02.gyptazy.de)	plb_anti_affinity_ntp

These rules leverage Proxmox's integrated access management, ensuring users can only define and manage rules for guests they have permission to access. The tag-based system maintains a streamlined approach while preserving Proxmox's permission model.





Maintenance Mode

ß (~) $\triangleright | \triangleleft$

Automatic Migration

available nodes in the cluster

Perform Maintenance

without disruption

Designate Nodes

Add nodes to the maintenance_nodes list in your configuration file

Return to Service

Remove from maintenance list and restart ProxLB to resume normal operation

When a node is set to maintenance, no new guest workloads will be assigned to it, and all existing workloads will be migrated to other available nodes within the cluster. This process ensures that affinity rules and resource availability are respected, preventing disruptions while maintaining optimal performance across the infrastructure.

- All workloads are moved to other

- Update, reboot, or service the node



Support and Community



GitHub

Report bugs, contribute code, or access documentation through the official GitHub repository. The project is open-source under GPL-3.0 license with 8 contributors and regular updates.



Matrix & Discord

Join the dedicated chat rooms on Matrix (#proxlb:gyptazy.com) or Discord for realtime support and community discussions. Both platforms are bridged to ensure unified communication.



Documentation

Access comprehensive documentation in the GitHub repository's docs directory, covering installation, configuration, and advanced usage scenarios for different deployment methods.

ProxLB is maintained by Florian Paul Azim Hoberg (@gyptazy) as a one-person project that includes coding, testing, packaging, and infrastructure management. The community's support through bug reports, feature requests, and contributions helps drive the project forward.





About the Maintainer: gyptazy

ProxLB is developed & maintained by Florian Paul Azim Hoberg (gyptazy), a passionate DevOps engineer bridging development and operations.

As the sole maintainer, gyptazy handles all aspects of ProxLB: coding, testing, packaging, and infrastructure management.



Contact

Email: gyptazy@gyptazy.com

Mastodon:

Matrix:



Social

- @gyptazy@gyptazy.com
- @gyptazy:gyptazy.com



🙆 Made with Gamma