BoxyBSD

Creating a free VPS hosting platform for beginners and newcomers to learn and educate themselves into the BSD world!

Welcome to BoxyBSD!

A non-commercial project by gyptazy

Who's gyptazy?

Florian Paul Azim Hoberg

- Known as **gyptazy** in the tech community
- Working at **credativ GmbH** in Mönchengladbach, Germany
- Technical Lead / Sr. Consultant
- FOSS Contributor
- Loves FreeBSD

Garden Linux

Developer & Maintainer at the Garden Linux project. Mostly working on ARM64, SELinux and PyTest.

Ansible Modules

Developing & contributing powerful Ansible modules for infrastructure management.

ProxLB

Founder and lead developer of this innovative load balancing solution for Proxmox environments.

BoxyBSD

Founder & lead developer of the BoxyBSD project. Providing free VPS instances to newcomers & beginners to learn and practice BSD based system.

What is BoxyBSD?

BoxyBSD is basically a VPS hosting provider - just free!

But it's more than just a typical provider.

Gateway to the BSD world

A free gateway to easily get in touch with BSD based systems.

- No own hardware required
- Ready2Use Images or self-install
- Multiple BSD variants *(FreeBSD, OpenBSD, NetBSD,...)*
- Fully automated *(inc. self-service portal)*
- Community *(cooperating with BSD Cafe)*

Educational Focus

Designed specifically for beginners and newcomers to explore and learn BSD-based operating systems.

- Free *(eliminating payment barriers)*
- Learning by Doing
- Bridging theory and practice
- Open to everyone
- A place to start with BSD

- *Cooperating with schools & universities in Africa & India*

Technically

Providing accessible, no-cost VPS instances to eliminate financial barriers for learning.

- No personal data collected
- Dedicated VPS instances
- Multiple GEO locations
- Running within virtualization
- Static & public IP addresses

Why BoxyBSD?

BoxyBSD is driven by a strong commitment to give back to the open-source community, foster education, and strengthen the BSD ecosystem.

Returning to the Community

We believe in the spirit of open-source and aim to contribute meaningfully by providing accessible resources for everyone.

Empowering Learning & Education

Our platform serves as a hands-on environment for newcomers and students to explore and master BSD-based operating systems.

Boosting the BSD Ecosystem

By lowering barriers to entry, we encourage more users and developers to engage with and contribute to the broader BSD community.

Strengthen the Open-Source Community

Today's beginners might be the leading developers in future, brining open-source based solutions a step ahead!

Can you guess which Hypervisor drives BoxyBSD?

You'll probably be wondering...

History of BoxyBSD

From humble beginnings to a thriving community platform, BoxyBSD has evolved significantly.

1 Founded in late 2022

- Initial project ideas
- Running on spare-hardware of gyptazy

2 Operating as a "Network Debugging Service"

- First public alpha portal
- FreeBSD Jail based
- Spawning ad-hoc instances for debugging

3 Switching to VMs in 2023

- Switching to bhyve hypervisor (on FreeBSD)
- Instances have been semi-automated deployed
  - Resulting in high pressure on delivery
- Switching target user base to beginners & newcomers

4 Fully-automated Deployments in mid. 2023

- Switching from bhyve to KVM *(via Proxmox VE)*
- Getting sponsors who share the passion
  - *(Nerdscave Hosting, Macarne, Servermanagementpanel)*
- Creating a web based control-panel for users
- Creating a fully-automated pipeline
  - Crafted ProxLB
  - Crafted several Proxmox Ansible Modules

5 BoxyBSD 2.0 Portal in late 2025

- Shipped more than 1k VPS boxes
- Added DN42 BGP Peering
- Added multiple regions over the world
- Created a new & modern control-panel for users
- Learned a lot myself
  - Including much fun of abuse and anti-abuse
  - Met several great people, creating awesome connections
  - Still a one-man show

Why Switching the hypervisor?

Why did I move from bhyve to KVM (Proxmox VE)? First, let's have a look at the requirements of operating such a platform.

Requirements for Operating Such a Platform

When choosing the right hypervisor and infrastructure for BoxyBSD, several critical requirements guided our decision-making process to ensure stability, scalability, and ease of use.

Fully Automatable

The platform needed to support comprehensive automation for provisioning, management, and scaling of VPS instances.

Hardware Agnostic (x86_64)

Compatibility with standard x86_64 hardware was essential for flexibility and cost-effectiveness across diverse server environments, especially when sponsors come up with any kind of spare hardware.

Robust Storage Support

Independent and advanced storage solutions, particularly ZFS, were crucial for data integrity, snapshots, and flexible storage management.

Clustering Capabilities

The ability to pool resources and manage multiple nodes as a single system was a key requirement for scalability and distributed operations.

High-Availability (HA)

Built-in features for high availability were necessary to ensure continuous service and minimize downtime for user instances.

Live Migrations

The capacity for live migration of virtual machines between hosts was critical for maintenance, upgrades, and load balancing without service interruption. No ones wants VPS downtimes because of installing security updates on the hypervisor nodes.

Long-Term Maintainability

Ease of maintenance, updates, and troubleshooting over an extended period was paramount for sustainable operations.

Overview: bhyve Hypervisor

Before diving into why we transitioned, let's first understand the core characteristics of bhyve, a powerful yet distinct hypervisor.

Secure & Fast Performance

bhyve offers excellent security features and near-native performance for virtual machines, making it a robust choice for isolation and efficiency.

Easy to Maintain

Its integration with the FreeBSD kernel simplifies system management, allowing for straightforward updates and maintenance within a familiar environment.

CLI-Driven Orchestration

Many command-line based orchestrators are available, providing powerful and flexible control for advanced users and scripting.

Limited UI/API Availability

While powerful, user interface and extensive API-based orchestration solutions for bhyve are comparatively rare, presenting a unique challenge for broader adoption.

The limited UI/API availability for bhyve was a significant factor we'll explore further when discussing the hypervisor switch.

Overview: KVM Hypervisor

Kernel-based Virtual Machine (KVM) is a robust and widely adopted open-source virtualization technology built into Linux. It's known for its flexibility and strong performance.

Secure & Fast

KVM delivers robust security and near-native performance, crucial for demanding virtualized workloads.

Easy to Manage

Thanks to its integration with the Linux kernel, KVM is generally straightforward to deploy, configure, and maintain.

CLI-Driven Orchestration

A rich ecosystem of command-line tools provides powerful and flexible orchestration capabilities for automation.

Rich UI/API Ecosystem

KVM benefits from numerous advanced UI and API solutions, like Proxmox VE, enabling sophisticated management and automation.

Overview: Other Solutions

During our evaluation, we also considered and tested various other virtualization platforms such as Proxmox VE, XCP-ng, Harvester, and Incus. However, they presented challenges that ultimately led me to pursue different options.

Hardware Flexibility

Some solutions struggled to efficiently manage the diverse range of node memory capacities, starting from 32GB up to 512GB, where the whole solution already required a minimum of 32GB memory.

Driver & Architecture Compatibility

Issues with microkernel architectures and a lack of critical drivers prevented seamless integration with our existing and sponsored hardware.

Automation Limitations

A significant drawback was the absence of robust API support, preventing the ability to fully automate provisioning and management processes or inefficient resource schedulers.

Why Switching the Solution?

The transition from bhyve to KVM/Proxmox was a strategic move to overcome early challenges and scale BoxyBSD effectively.

bhyve Limitations

- Missing Orchestrator solution with great API support [*]
- Missing Clustering support [*]
- Missing Live-Migration of VMs [*]

KVM/Proxmox Adoption

- Great API for further automation
- Clustering
- Live-Migration
  - VM & Storage

The limited UI/API availability for bhyve (of any orchestrator) was a significant factor and we'll explore the current situation later.

Ok, Proxmox - but...

While Linux with KVM/Proxmox is a great foundation, the journey wasn't without its own set of unique challenges that required custom solutions and deep dives into infrastructure management.

Missing/Faulty Ansible Modules

Integration with infrastructure-as-code tools like Ansible proved challenging due to immature or incomplete modules, necessitating custom scripting.

Outcome: Developed several modules and brought the upstream. (`e.g. proxmox_cluster`, `proxmox_node`, `proxmox_storage`, among others)

Missing Real Cluster Resource Scheduler

Proxmox lacks an advanced Distributed Resource Scheduler (DRS) compared to other solutions, requiring manual optimization.

Outcome: Developed the new project **ProxLB**.

Missing Real Micro-Segmentation Solution

Missing an integrated micro-segmentation solution for fine-grained network security is absent, demanding external firewalls or network virtualization overlays.

Outcome: Developed the new project **ProxWall**.

Multi-Master with Corosync/PMXCFS Can Be a Mess

Managing multi-master configurations, especially with Corosync and PMXCFS, can be complex and prone to issues without careful planning and expertise - especially limiting cluster sizes and stretched- or multi region clusters by latency.

Outcome: Dealing with it by creating layer-2 VPNs without HA and managing them through self-written solutions.

Missing Global Cluster Management

Scaling across multiple, geographically dispersed clusters lacks a centralized global management interface, complicating large-scale deployments.

Outcome: Proxmox Datacenter Manager provides a limited overview but no real actions. Further useful API automation PRs were rejected, so I manage this externally.

VPS Self-Service Portal

There are already existing solutions for VPS hosting providers to offer self-services to end-users. However, most of them are not open-source (trust), expensive or over-sized.

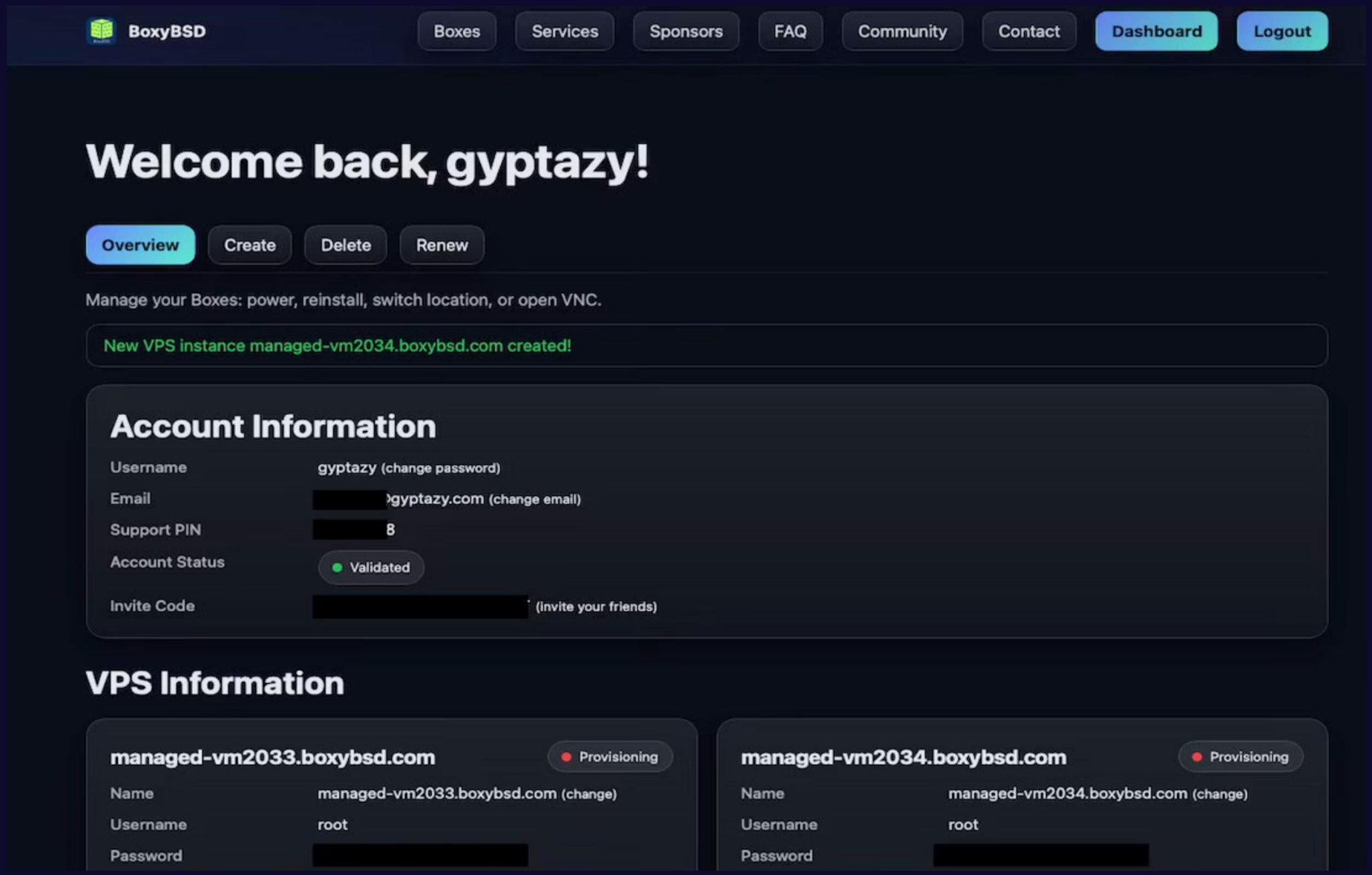
Outcome: Creating a whole self-service portal including front- & backend based on Proxmox API, Ansible & Terraform.

Running BoxyBSD - All over the World!

Having all required toolings developed, I'm able to operate clusters worldwide with a minimum of efforts. Currently operating in:

- Canada, Toronto
- France, Roubaix
- Germany, Frankfurt
- Germany, Limburg
- Italy, Milan
- Japan, Tokyo
- Netherlands, Amsterdam
- Netherlands, Eygelshoven
- Singapore, Singapore
- Ukraine, Kyiv
- USA, New York

The Current Self-Service Portal



Proxmox: Conclusion

Solid Foundation

Proxmox offers a robust and reliable foundation for many virtualization needs, by using the power of KVM.

Feature-Rich Core

It already comes out of the box with many great features for storage- & user management, including the API.

Enterprise Gaps

While powerful, some advanced enterprise features, like a fully integrated DRS, DPM, micro-segmentation or a real global cluster management, are not natively present.

Custom Solutions

These gaps can be effectively filled with third-party tools or custom-developed solutions like ProxLB and ProxWall.

Development Agility

The open-source nature and robust API allow for easy development and integration of missing functionalities.

Future-Proofing

It's definitely worth keeping an eye on other solutions like Harvester, Incus or Sylve.

One more thing

SYLVE

Data Center

Data Center

2025: FreeBSD, bhyve - Sylve is coming!

A new era for bhyve virtualization is on the horizon with the upcoming Sylve project, set to bring advanced management capabilities and a robust ecosystem to FreeBSD users.

Proxmox-Inspired Web-UI

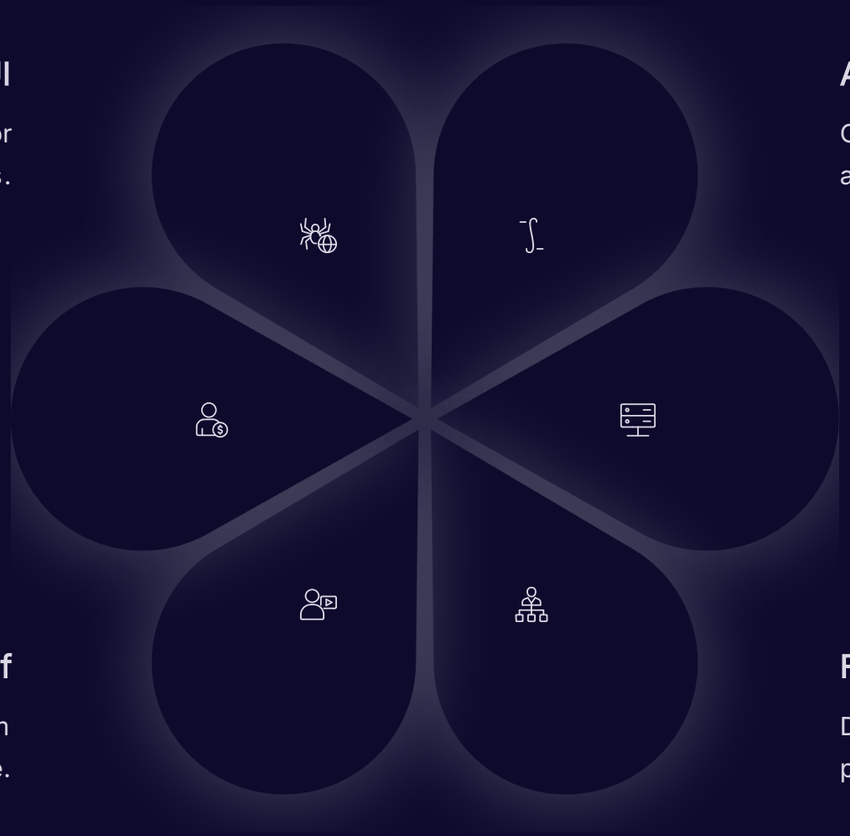
Providing an intuitive and powerful web-based interface for effortless management of virtual machines.

Sponsored by FreeBSD Foundation & Alchemilla

Supported by key organizations dedicated to advancing open-source technologies and the FreeBSD ecosystem.

Lead Developer: Hayzam Sherif

Driven by the vision and expertise of lead developer Hayzam Sherif, bringing years of virtualization experience.



API Integration for Automation

Offering comprehensive API support to enable seamless automation and orchestration within existing workflows.

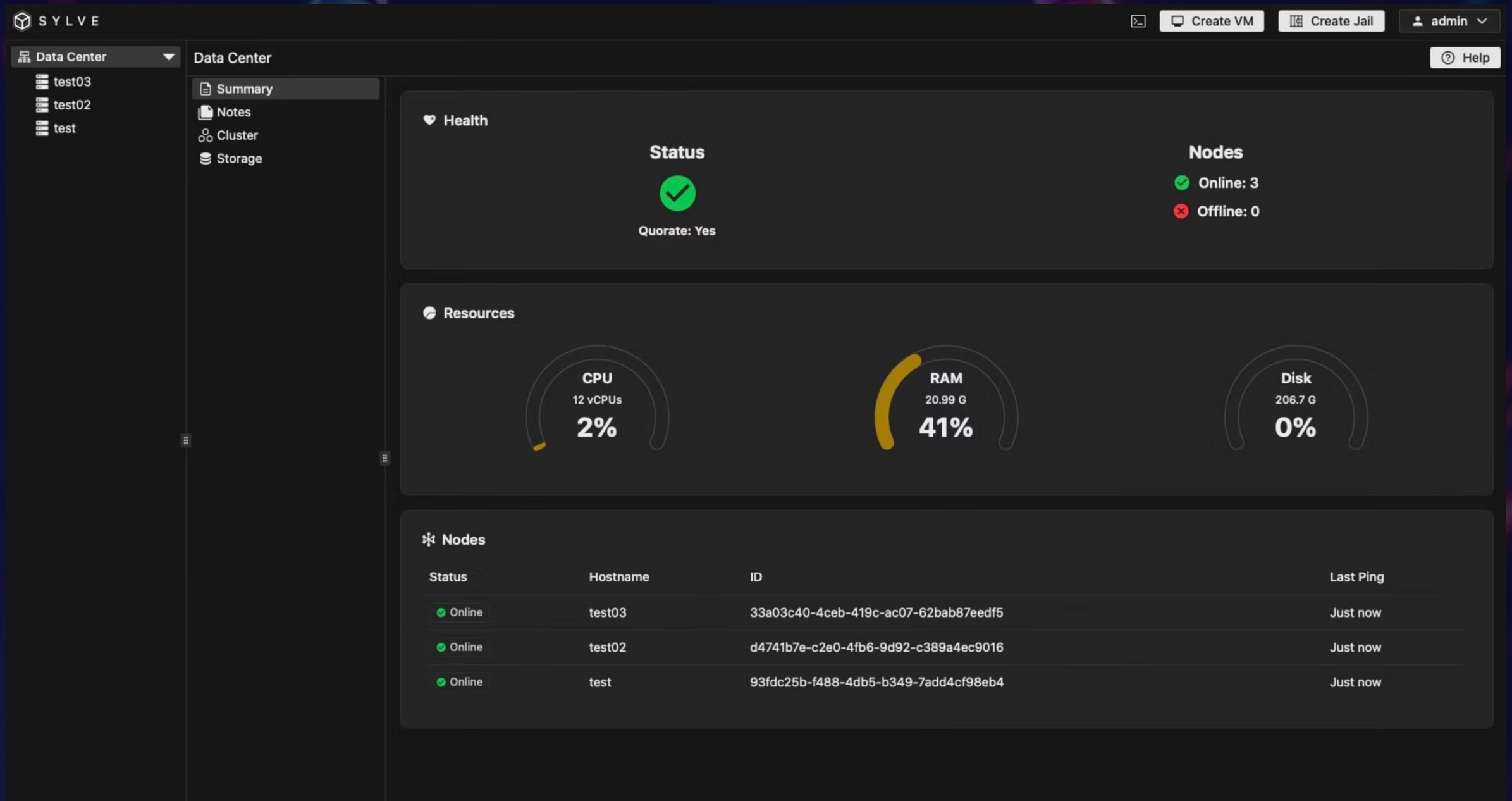
Clustering Support

Introducing native clustering capabilities for high availability and scalable virtualization deployments.

FreeBSD First Approach

Designed with a strong focus on FreeBSD, ensuring optimal performance and integration with the operating system.

Sylve on FreeBSD - A Proxmox alike UI



Special Thanks & Greets

- **Alexander Wirt** (@formorer, credativ GmbH)
- **Arne** (Macarne Ltd.)
- **Deb Godwin** (FreeBSD Foundation)
- **Emile "iMil" Heitor** (@imil, SmolBSD / NetBSD)
- **Hayzam Sherif** (@Hayzam, Alchemilla)
- **Michael Dexter** (@dexter, FreeBSD / bhyve Calls)
- **Moritz Mantel** (@MinersWin, Nerdscave-Hosting)
- **Stefano Marinelli** (@Stefano, BSD.Cafe)

And of course all other ones which I couldn't credit all here...!

Questions?

Thanks!

Author: Florian Paul Azim Hoberg @gyptazy

Web:	https://gyptazy.com
Email:	gyptazy@gyptazy.com (0xB1B88CBB)
Matrix:	@gyptazy:gyptazy.com
X:	@gyptazy
Fediverse:	@gyptazy@gyptazy.com