Canonical

OpenStack made easy

February 2023

Executive summary

OpenStack remains the world's leading open-source cloud platform and its adoption continues to grow every year [1]. According to the latest results from an annual OpenStack User Survey, published in November 2022, OpenStack now powers more than 40 million cores in production across over 300 surveyed deployments [2]. Not to mention thousands of other deployments running all over the world not documented anywhere.

This phenomenon has its roots in the superior economic advantages that OpenStack brings to organisations of all sizes and across industries. Being an open-source project, OpenStack allows for significant cost savings compared to proprietary virtualisation solutions, such as those in VMware, Citrix and Proxmox families. At the same time, it is a fully functional cloud platform which serves as an extension or a reasonable alternative to hyperscalers, effectively addressing cloud cost optimisation and digital sovereignty concerns [3].

At the same time, OpenStack adoption has always been a challenge. This is certainly the case for those with no previous experience with Linux and cloud computing. But even the most experienced organisations can struggle to implement OpenStack. Thousands of available design options, excessive hardware requirements and complex installation procedures in the upstream have been blocking newcomers from even trying it. Due to its versatility and scale, OpenStack has been pinned by the open source community as a complex project.

In this whitepaper, we demonstrate Canonical's ultimate approach to making OpenStack easy. We show that typical challenges with its adoption can be addressed by using sensible defaults and appropriate tools. Later, we discuss what those tools are, how they fit together and how OpenStack architecture looks when it runs the canonical way. The whitepaper wraps up with an overview of the installation process. To be precise, we demonstrate how to get OpenStack up and running on a workstation in no more than 20 minutes using our recommended tooling.

Contents

1. OpenStack challenges

Newcomers usually face several challenges when they first try OpenStack and they quickly get lost. This is very unfortunate. Thousands of OpenStack developers have been expanding its capabilities for years, making it a reasonable alternative to leading public clouds. Yet, only a few organisations have prioritised lowering the entry barrier for beginners. The following section provides a brief overview of the most typical challenges with OpenStack adoption and their impact on the learning curve.

1.1 Thousands of design options

One of the biggest advantages of OpenStack compared to other open-source cloud platforms is its versatility. For decades various organisations all over the world have been contributing patches and integrating OpenStack with their products. As a result, OpenStack supports all leading central processing unit (CPU) architectures, hypervisors, tens of software-defined networking (SDN) solutions and even more storage platforms. Its integration capabilities are almost unlimited.

Moreover, OpenStack uses modular architecture with each of its services being developed and maintained independently as a separate project. While some of those projects are stable and mature, others are still incubating or have been abandoned. In total, OpenStack consists of around 30 services, 80 configuration files and 5,000 configuration options. Its services can be co-located or distributed across the cluster. All of that results in thousands of possible design options, making OpenStack complete, but also hard to learn.

1.2 Dedicated hardware required

OpenStack is an infrastructure-as-a-service (IaaS) cloud platform, meaning that it enables provisioning of virtual machines (VMs) through a self-service portal. This also means, however, that OpenStack ideally should run on bare metal. This creates another challenge as spare hardware is not always available on site. To complicate things even further, most of the upstream installation instructions require at least several physical machines. Finally, since OpenStack is inherently huge and consumes a lot of resources by default, beefy hardware is often required to deliver an ultimate experience.

1.3 Complicated installation instructions

OpenStack installation is where most of its users usually struggle after they manage to make some initial design decisions and arrange the required hardware. This is because OpenStack installation instructions in the upstream are overly complicated. A typical procedure involves adding necessary software repositories, installing required packages, updating OpenStack configuration files, creating databases, populating them with sample records, etc. Even though they are worth executing step-by-step for learning purposes, the instructions are unnecessarily complex for those who want to test OpenStack during their lunch break.

1.4 Time-consuming operations

A successful installation of OpenStack is usually just the beginning of a bigger journey. While the first proof of concept (PoC) environment can be re-deployed on demand, the production environment has to be operated on a regular basis, sometimes for years. This creates another challenge. Historically, OpenStack operations used to be time-consuming. This is especially relevant in case of complex multi-step operations that have to be properly planned, such as OpenStack upgrades. Even though most organisations finally manage to deal with OpenStack installations, many have been struggling with its operations beyond day 1.

2. Simplifying OpenStack adoption

MicroStack is the most straightforward OpenStack ever. It installs in minutes anywhere through a friendly interface, making OpenStack fully accessible to newcomers. Thanks to its opinionated nature and container-based architecture, MicroStack effectively delivers distilled OpenStack excellence with a native Kubernetes experience.

OpenStack has and will always have its challenges. Those result directly from the nature of the project: it aims to provide a comprehensive, enterprise-grade and open-source cloud platform that ensures comparable functionalities to hyperscalers. However, by using the right architecture and proper tooling, it is possible to make OpenStack easy to use. This section highlights the Canonical way to straighten up OpenStack adoption.

2.1 Sensible defaults

We already know that OpenStack has thousands of possible design options. Its versatility makes it suitable for very specialised use cases, such as a local public cloud infrastructure or telco network function virtualisation infrastructure (NFVI). Not every organisation is or has an ambition to become a service provider, however. In fact, OpenStack is mostly used by small and mid-size enterprises, looking for a cost-effective general-purpose private cloud platform.

The latest results from the annual OpenStack User Survey clearly show that the vast majority of OpenStack deployments are based on some standard design options [4]. The leading choices for hypervisor, SDN solution and storage platform are Kernel-based Virtual Machine (KVM) / Quick EMUlator (QEMU), Open vSwitch (OVS) / Open VIrtual Network (OVN) and Ceph respectively. Moreover, only 22 out of 49 officially supported projects are actively used in 80% of reported production environments. Sticking to sensible defaults enables organisations to get their test deployment up and running quickly, while they can continue exploring all the capabilities later.

2.2 Minimal footprint

Even though production OpenStack environments require several beefy physical machines, arranging such hardware for testing purposes is not necessary. OpenStack can run in a single-node mode out of the box. However, even a single-node installation requires a lot of resources by default, making it hard to test without dedicated hardware. This is where various packaging mechanisms come into play. By using the right mechanism for each of its services and carefully optimising their parameters it is possible to minimise the footprint of OpenStack, making it installable on workstations or even VMs.

2.3 Lucid interface

While even the most complicated instructions can be fully automated using appropriate tools, some input from the user is still required when performing an OpenStack installation. This is because some settings, such as networking and identities need to be configured accordingly. That's why wrapping OpenStack up with a friendly interface makes a lot of sense. Users can answer some essential questions during the initial deployment and request certain post-deployment actions, while benefiting from total bottom-up automation taking place in the background.

2.4 K8s-native framework

Kubernetes (K8s) has undoubtedly become the de facto standard for hosting modern cloud applications. At the same time, many Kubernetes-based OpenStack platforms, such as the Airship project, have been struggling to get initial adoption in the field [5]. This is because not all OpenStack components are suitable for being containerised. Moreover, simply packaging OpenStack services inside open container initiative (OCI) images does not solve all the problems as Kubernetes does not address day-2 challenges on its own.

What all those legacy Kubernetes-based OpenStack platforms have been missing are Kubernetes operators [6]. Those software extensions encapsulate the knowledge, wisdom and expertise of real-world operations teams, and codify it into computer programs that help to operate complex application ecosystems. As a result, not only the initial deployment of OpenStack, but also its post-deployment operations become fully automated, while ensuring true K8s-native experience.

3. MicroStack architecture

MicroStack is a software implementation of all the aforementioned methods to straighten up OpenStack adoption. It uses sensible defaults, it installs anywhere through a lucid interface and ensures true Kubernetes-native experience. As a result, OpenStack becomes way easier and more accessible than it used to be. In the next section we lift the veil of secrecy and demonstrate what MicroStack architecture looks like and what its building blocks are.

3.1 Building blocks

MicroStack uses three primary types of artifacts for the initial deployment and its post-deployment operations: Those include snaps, OCI images and charmed operators. We will start with a brief overview of those three software packaging mechanisms. Later, we will show how they are used together to deliver OpenStack control plane and data plane services in MicroStack.

- ### Snaps
  Snaps are software packages that contain application code along with all its dependencies and libraries. They typically run in isolation from other installed applications, within a confined environment, delivering a high level of security. Snaps are cross-platform by nature and can be installed easily across all Linux distributions. They update automatically through a transaction-based mechanism, effectively making the underlying infrastructure immutable.

- ### OCI images
  OCI images are container images built according to the OCI standard [7]. They are based on the Docker manifest and are fully compatible with Kubernetes specifications. Even though Canonical maintains a repository of Long Term Supported (LTS) images for the most popular open-source applications, at the time of writing, MicroStack relies on the Kolla project to deliver OCI images for OpenStack services.

- ### Charmed operators
  Charmed operators are a new class of operators that enable reuse across a wide range of substrates. They package applications' operations code and drive the deployment, integration and operations of applications across bare metal machines, VMs and containers. As a result, charmed operators enable fully automated day-2 capabilities with minimal human intervention required.

3.2. Control plane

In MicroStack, the OpenStack control plane is entirely Kubernetes-based. This includes its databases, messaging queues, the OpenStack dashboard and a bunch of OpenStack services, such as Identity application programming interface (API) or Compute Scheduler. Each of those components runs as a separate pod with the primary container hosting the application code and the sidecar container hosting the operator code. MicroStack uses MicroK8s - a lightweight Kubernetes cluster - for container coordination purposes. MicroK8s gets installed on the host as a snap.

3.3. Data plane

In turn, the OpenStack data plane in MicroStack is snap-only. This is because data plane components, such as QEMU or OVS need to interact with the hardware directly. Therefore, MicroStack delivers all data plane components inside of a single snap - openstack-hypervisor - which gets installed directly on the host. Another snap - microstack - serves as a wrapper behind all control plane and data plane components, acting as an ultimate interface to the OpenStack cloud.

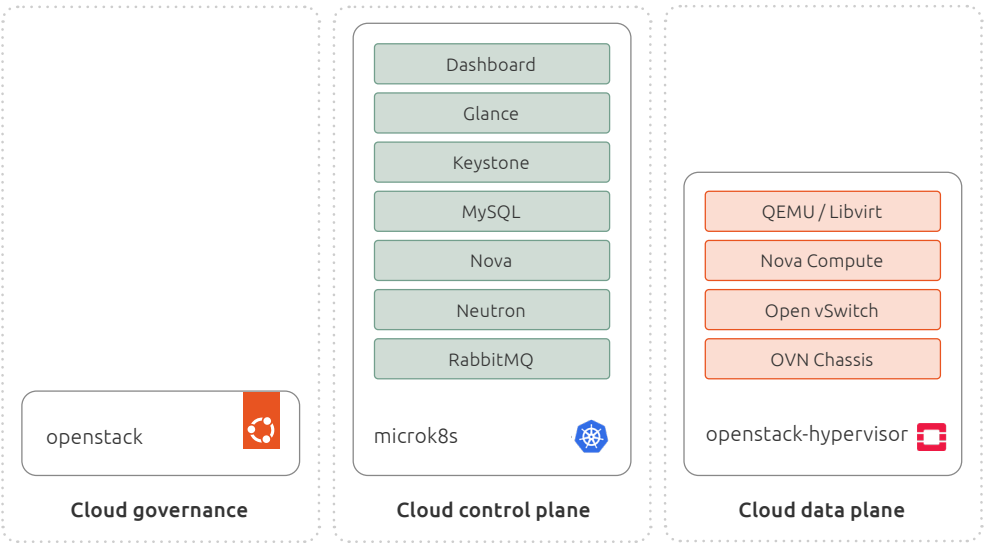

Fig. 1. MicroStack architecture

4. Breaking the ice with OpenStack

Now that we have explored MicroStack's architecture and its benefits, let's break the ice with OpenStack and get your PoC environment up and running. In the following steps, we will discuss what the MicroStack installation process looks like. Please refer to the official installation instructions for detailed information on how to get started with MicroStack or follow our series of tutorials for beginners for a more comprehensive experience.

4.1 Prepare

The easiest way to test MicroStack is to install it on your workstation or inside of a VM. Please note that the latter option requires nested virtualisation to be enabled first. MicroStack requires a multi-core amd64 CPU, 8 GB or random-access memory (RAM) and 100 GB of storage. Even though it works across all Linux distributions, Canonical tests MicroStack on a regular basis on the latest Ubuntu LTS versions. Windows and MacOS users can install MicroStack through Multipass by getting an Ubuntu VM running on their workstation quickly.

4.2 Install

During this phase you install all necessary components to initialise MicroStack. Those include micro8s, openstack-hypervisor and microstack snaps. You also enable all required extensions to your Kubernetes cluster, such as storage or load balancing that are not enabled in MicroK8s by default. Finally, you adjust permissions to the microk8s snap to be able to bootstrap the OpenStack control plane on top of the Kubernetes cluster.

4.3 Initialise

MicroStack can be initialised in various modes. In the simplest, single-node mode it spins up one unit of each control plane service on MicroK8s. It uses OCI images and charmed operators to get all services up and running and then connects the entire control plane to the data plane services installed by the openstack-hypervisor snap. Later, MicroStack can be initially configured to set up networking and create all necessary records, such as identities and images - everything that's needed to launch your first instances.

4.4 Interact

At this point, your OpenStack is ready for use. You can interact with it in multiple ways. First of all, you can use the microstack command for cluster administration purposes. You can also install an OpenStack client and extract OpenStack RC files from your MicroStack installation to be able to interact with the cloud through a command line interface (CLI). Finally, you can use the OpenStack dashboard to create all types of records, including user accounts, projects, virtual networks and cloud instances.

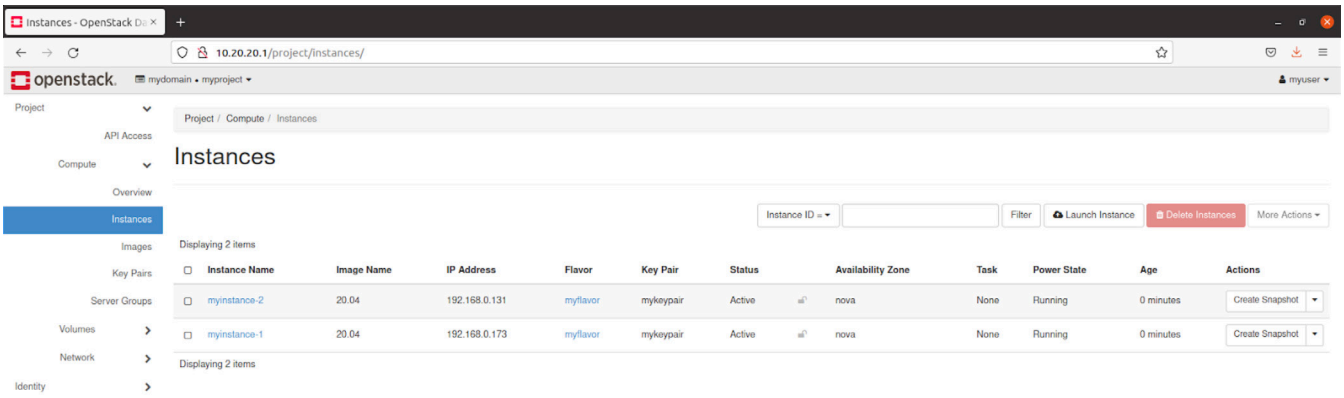

Fig. 2. OpenStack dashboard

5. Running OpenStack in production with confidence

Getting the PoC up and running is usually the first step to a bigger project. The beauty of MicroStack is that it enables OpenStack deployments on any scale: from single-node installations to large-scale clusters. In turn, the beauty of Ubuntu is that organisations can use it free-of-charge or with an optional, paid support subscription. There are also many other commercial services available from Canonical for enterprise customers. Those include:

- **Private Cloud Build (PCB) and PCB Plus** – Fixed-price consultancy packages for OpenStack implementation on reference architecture and certified hardware, including cloud design and delivery, on-prem workshops, workload analysis and migration plans.
- **Ubuntu Pro** – Enterprise subscription for Ubuntu that covers all layers of the infrastructure, including 10 years of security updates, phone and ticket support, production-grade service level agreements (SLAs) and regulatory compliance programmes.
- **Managed OpenStack** – Fully-managed cloud service, including cloud monitoring, maintenance, daily operations, incident and problem resolution, software updates and OpenStack upgrades that enable organisations to fully outsource the management of their OpenStack.

6. Conclusions

OpenStack adoption usually entails several challenges which result directly from the nature of the project. Being the world's leading open-source cloud platform, OpenStack is massive and relatively complex. Newcomers often get lost in the rich diversity of its configuration options, supported projects and actual system requirements. As a result, even very simple tasks, such as getting it up and running for testing or PoC purposes, used to be nontrivial for years as they required going through intricate installation procedures. Not to mention its post-deployment operations when in production.

MicroStack was designed to address those challenges and make OpenStack fully accessible to people with no previous experience in cloud computing. Its sensible defaults and lucid interface enable engineers to experiment with OpenStack immediately without spending hours on figuring out design decisions and complex upstream installation procedures. Its K8s-native interface makes it very intuitive for those with Kubernetes background. Finally, its minimal footprint makes it suitable for devices with limited hardware resources, including developer workstations. This enables newcomers to try OpenStack in a single-node mode, while using the same artifacts to deploy it and operate in production on a large scale.

Learn more

- Try OpenStack today. Canonical provides the most straightforward installation instructions for OpenStack ever. All you need is your Ubuntu-based workstation.

- Learn OpenStack through a series of tutorials for beginners. Explore how to use OpenStack for cloud infrastructure implementation purposes, from a single-node installation to large-scale clusters.

- Visit our page to learn more about Canonical's OpenStack offering and commercial services available to enterprise customers.

- Get in touch with Canonical experts.

References

1. https://ubuntu.com/engage/openstack-introduction
2. https://www.openstack.org/user-survey/2022-user-survey-report
3. https://www.brighttalk.com/webcast/6793/517097
4. https://www.openstack.org/analytics/
5. https://www.airshipit.org/
6. https://ubuntu.com/engage/kubernetes-operators-explained-whitepaper
7. https://opencontainers.org/

Canonical Ubuntu