

To virtualize or not to virtualize:
A comparison between self-hosted virtualization using Proxmox
and using the cloud

David Cederström
Arasp Keighobad

April 24, 2023

DD2482

We certify that generative AI, incl. ChatGPT, has not been used to write this essay. Using
generative AI without permission is considered academic misconduct

1 Introduction

In today's rapidly evolving technology landscape, organizations and companies need to be agile and responsive in order to be able to adapt according to novel business needs. DevOps has emerged as a popular umbrella term for all the different methodologies and strategies leveraged in the software development cycle. It encompasses tools, practices and cultural philosophies that enable organizations to quickly and effectively deliver applications and services at a high rate, maximizing the provided value of their products. Virtualization software and cloud platforms play a crucial part in DevOps by providing a flexible and scalable infrastructure for deploying, managing and automating the applications.

Using some type of virtualization software, such as Proxmox, VMWare, or Hyper-V, enables organizations to create and manage multiple virtual machines (VMs) on a single physical server, which is a very efficient use of hardware resources.[10] Since virtualization also allows you to easily replicate existing environments and to rapidly deploy new VMs, it is a great tool for DevOps teams that quickly might need to launch new instances for testing and development.

Cloud platforms such as Amazon Web Services (AWS) and Microsoft Azure (Azure) takes virtualization one step further by offering a subscription-based service that allows for access to networked storage space and computing resources.[2],[3] This allows organizations to quickly scale their infrastructure in case of changing demands. Cloud platforms such as AWS and Azure also offer a wide range of managed services that can improve and be integrated with the DevOps process, with tools like automated deployment pipelines and serverless computing.

Since there is a vast amount of information about either method, it can be a difficult problem to know which fits one's needs. This essay will explore the pros and cons of using self-hosted virtualization software such as Proxmox compared to cloud-based virtualization platforms such as AWS and Azure. By examining the strengths and limitations of each approach, a better understanding of the best ways of deploying and managing infrastructure in a DevOps context will be provided.

2 Virtualization, Virtual Machines and hypervisors

Virtualization is the process of using software to create an abstraction layer over computer hardware, allowing a single computer to be divided into multiple so called virtual machines (VMs). The VMs themselves, have their own operating systems making them similar to independent computers, however, only using a small part of the actual hardware, specifically allocated to the VM by the hypervisor. This could for example be allotted CPU cycles, memory and storage resources. This enables users to make more efficient use of their physical hardware. The software that creates and runs the VMs, is called a hypervisor, which operates as an interface between the hardware and the VM. The hypervisor is what makes sure that the VMs all have access to the parts of the hardware that they require in order to operate, as well as handling so that they work independently of each other.[1] There exist two types of hypervisors (also shown in figure 1):

- **Type 1**

This type of hypervisor is often also called *bare metal* hypervisors since they run directly on the physical hardware without needing an underlying operating system (OS).[1]

- **Type 2**

Unlike bare metal hypervisors, Type 2 hypervisors run as an application on an existing OS.[1]

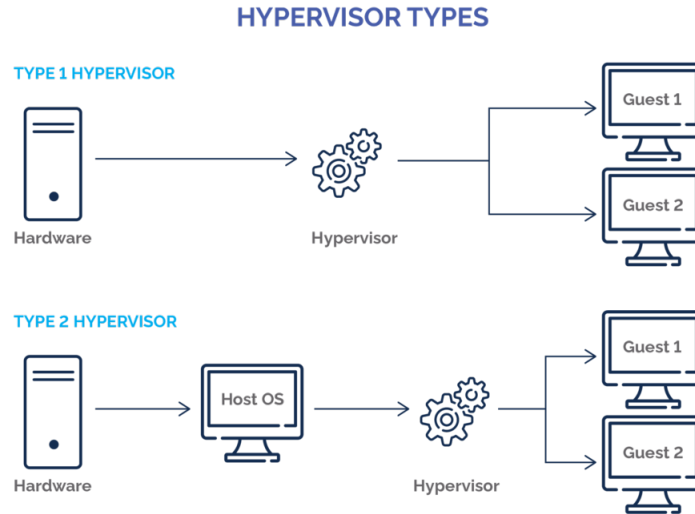


Figure 1: Overview of Type 1 (bare metal) and Type 2 hypervisors[11]

3 Self-Hosted Virtualization using Proxmox

3.1 Self-hosted virtualization

Self-hosted virtualization simply means that you run multiple VMs on a physical server, or a cluster of servers using virtualization software such as Proxmox, VMWare, or Hyper-V. This gives you more control over your infrastructure and allows you to tailor your virtualization to meet your specific needs. One could for example choose which hypervisor to use, how to allocate resources such as CPU and RAM in the preferred way, and which VMs to deploy. This can be an important aspect for organizations that have some certain or complex requirements or specific security measures that must be taken. Such as organizations within healthcare.[5]

3.2 Proxmox and its features

Proxmox Virtual Environment (VE) is a hosted server virtualization hypervisor that is open-source and designed to manage two virtualization technologies: Kernel-based Virtual Machine (KVM) for VMs, and Linux Containers (LXC) for containerization. KVM allows you to run multiple operating systems on a single physical machine without having to modify the guest OS while LXC provides the OS level virtualization. This means that it allows multiple isolated Linux systems to run on the same OS kernel.[9]

Proxmox VE is a so called Type 1 hypervisor, just as Hyper-V and VMware ESXi, which are two additional, popular hypervisors.[4] This means that they run directly on the host's hardware, as shown in figure 1 above. Proxmox VE also has an arsenal of integrated tools that enables easy configuration of high availability between servers, networking, software-defined storage, and disaster recovery, thereby providing a comprehensive virtualization solution for DevOps. One important feature that Proxmox supports is live migrations, which makes it possible for you to move a VM between hosts while its running, without having any downtime. Furthermore, Proxmox VE's web-based interface (which can be seen in figure 2 below) enables efficient management by combining various functions into one platform.[9]

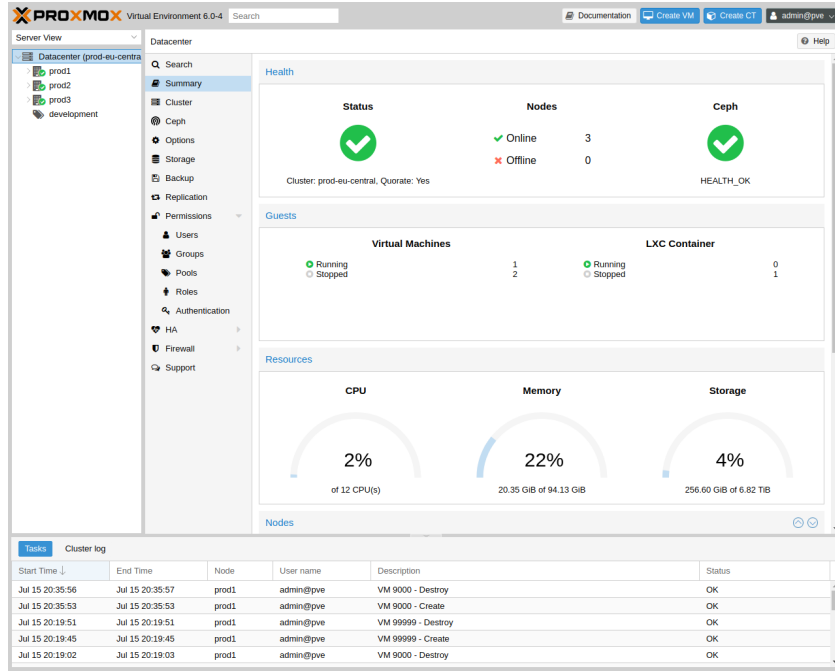


Figure 2: Graphical User Interface (GUI) of Proxmox where users can get information regarding statuses of processes, configurations, log entries, setup etc [11]

4 Cloud-Based hosting using AWS, Azure or GCP

Cloud based virtualization and hosting entails using pre-arranged services on platforms running in the data centers and hardware of other companies. The three services that currently have the biggest market share are Amazon Web Services (AWS), Azure (Microsoft) and the Google Cloud Platform (GCP).[15] Each come with their specific strengths and weaknesses, however, most notably, one does not have to set up, host and maintain actual hardware. This allows for more predictable costs, as well as higher guarantees of up-time and performance with respect to geographical location of users since these cloud services tend to have a very large amount of data centers spread out around the world, providing fast and continuous service.[7]

All the different cloud providers more or less by definition have to provide the same basic functionality categories of "compute and storage", in other words, that is the different types of services provided that meet the computational and storage needs of the customers' applications. On the other hand, they all differ slightly on specific details.

One important difference considered by many is pricing. Since all the platforms provide complex pricing structures, with different final costs based on what specific services are used, for how long, and how often, it is difficult to make a general one-to-one comparison between the alternatives.

- **AWS**

Based on reputation, AWS has a more complex pricing structure that can often be hard to understand for people new to AWS.[7]

- **Azure**

Azure offers a more straight-forward scheme with discounts based on long-term commitments.[7]

- **GCP**

GCP has a reputation of being the most cost-effective. Especially if one spends the time to optimize their use-case for their pricing structure.[7]

The services provided are another important component, and often, one uses ready made packaged solutions for services from the provider. AWS is known to provide a huge range covering most needs,

however, with the drawback of possibly being overwhelming for a new user. Azure comes in second with regards to number of services offered, however, it is also better integrated with services that are closely tied to Microsoft, such as Windows server and Office365. GCP is more focused and better tailored for developers and startups, as well as being concentrated on providing a good environment for open source software.

In conclusion, one can say that the differences are not huge in base functionality, rather in ease of use and familiarity, cost, and other mere details. One thing they all have in common, however, is enabling companies to not have the need of on-premise data centers in order to provide their service fast and reliably.

5 Comparison

Whereas the cloud platforms in essence all provide the same type of functionality, albeit with differences in some details, one can choose instead to use on-premises hosting. This comes with its own set of benefits and drawbacks, for which a trade-off has to be made to whether it is really worth it for the use-case in question to use either cloud-based solutions, or directly with the hypervisor using things like Proxmox.

The main concern about self-hosting, using software such as Proxmox, is the fact that one has a single point of failure. Measures that relate to the fact that one is maintaining their own hardware have to be taken to mitigate this, for example backups of the data used by the services as well as the server and its configuration itself. Preferably this is done with remote backups where the backups are not stored at the same location as the server, to mitigate unforeseen physical accidents.[12]

Another disadvantage of self hosting is the fact that it does not scale well. In the case where there is a sudden surge in demand for the service being provided, if the capabilities of the server do not meet the increased demand, one has to scale up in physical hardware. Which naturally takes time and money, at which point the service might have been unsatisfactory for the users over a prolonged period. Moreover, this same issue could lead to unused resources, in the case of periods of lower demand, leading to unnecessary energy and maintenance costs.[12]

On the other hand, if one wants to avoid those concerns, choosing the cloud based alternative means very easy and fast set-up without the need of hardware-specific considerations, as well as easy scaling as many of the platforms can quickly allocate additional resources to your service instance when demand goes up. Moreover, the reduced complexity of the architecture of your application that follows could mean reduced maintenance needs, needs for hardware upgrades and data center personnel, see figure 3.[12]

However, choosing the cloud alternative is at the cost of the better configuration abilities that are possible when running an on-premise instance. Specific configurations of the virtualization hypervisor that are in general not available to users of cloud services can be tweaked specifically to cater for the needs of one's own application to bring greater efficiency and performance.[6]

One other possible consideration that could make the use of cloud platforms impossible is security. The fact that there are some sensitive services handling confidential data means that one would not be able to store that data on other servers. Ways to mitigate this would then be to for example encrypting the data, which would add complexity to the system, as well as some performance hits.[14] There has for example been some recent cases where sensitive services related to medical care hosted in the European Union are not allowed to have data hosted on servers abroad (most often the cloud providers are companies based in USA). Such as the case where sensitive data from the Swedish Transport Agency were stored on IBM servers abroad. [13] Other examples could be military uses, sensitive governmental information, or trade secrets that need to be kept extra guarded.

One valuable thing gained from using many of the cloud platforms mentioned is the ability to get advanced support specific to one's own instance, reducing the need for hiring in-house consultants

and developers. Different plans in the different cloud platform services have high-quality on-demand professional service, which is highly valued in companies and organizations with tight uptime and performance requirements.

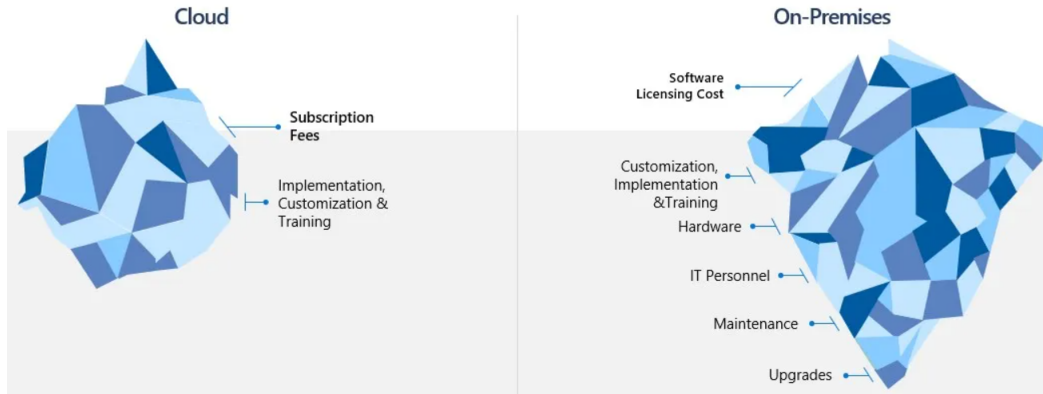


Figure 3: Illustrating the difference in work that needs to be done for a cloud implementation vs. on-premises [8]

6 Conclusion

Virtualization has become an essential part of today's modern software and IT infrastructure, especially in the context of DevOps. By using virtualization, DevOps teams are able to become more efficient by automating crucial processes of the development such as testing and delivery. This also leads to a better utilization of computer resources available, resulting in a quicker and more reliable delivery of software.

Main takeaway: Whether one should choose self-hosted virtualization, or utilize one of the cloud platforms discussed, boils down to one's needs and the circumstances of the specific problem at hand. One has to carefully consider things like cost, specific customization needs of the set-up, security, plans for longevity and scalability, support demands, and so on in order to know which of the solutions is right.

6.1 Reflection

Reflecting on the research process for this essay, we found that there is a lot of information about both virtualization and cloud computing available. This made it a bit challenging to narrow down the field to a specific aspect of virtualization. However, by digging deeper into the topic, we gained a better understanding of the technologies and concepts involved such as Proxmox, as well as strengths and limitations to these in the context of DevOps. Overall, it has been a valuable learning experience and we hope that our essay can help others when making decisions choosing between self-hosted and cloud-based virtualization platforms.

References

- [1] International Business Machines Corporation (IBM). “What is virtualization?” In: (2023). URL: <https://www.ibm.com/topics/virtualization>.
- [2] Amazon AWS. “What is AWS”. In: (2023). URL: <https://aws.amazon.com/what-is-aws/>.
- [3] Microsoft Azure. “What is Azure?” In: (2023). URL: <https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-azure/>.
- [4] N. Kraljević B. Đorđević V. Timčenko and N. Jovičić. “Performance comparison of KVM and Proxmox Type-1 Hypervisors”. In: (2022). URL: <https://ieeexplore.ieee.org/document/9983666>.
- [5] Rebecca Bernsen. “Why to self-host your software? Should you go for on-premises or cloud?” In: (2022). URL: <https://www.openproject.org/blog/why-self-hosting-software/>.
- [6] IBM cloud. “Bare Metal with a Hypervisor vs. Dedicated Host: What’s the difference?” In: (2022). URL: <https://www.youtube.com/watch?v=3xH1SAWd10>.
- [7] iLyas in the cloud. “Cloud Providers Compared: A Comprehensive Guide to AWS, Azure, and GCP”. In: (2023). URL: <https://www.youtube.com/watch?v=4AVQchL9tTo>.
- [8] DrWare. “Confront the misconceptions slowing your move to Dynamics 365”. In: (2021). URL: <https://www.drware.com/confront-the-misconceptions-slowing-your-move-to-dynamics-365/>.
- [9] Proxmox Server Solutions GmbH. “Features”. In: (). URL: <https://www.proxmox.com/en/proxmox-ve/features>.
- [10] Rik Goldman. “Learning Proxmox”. In: (2016), p. 11. URL: https://books.google.se/books?hl=sv&lr=&id=nrljCwAAQBAJ&oi=fnd&pg=PP1&dq=proxmox&ots=sxGj2i_a_G&sig=W49AgftQtB2ET6IPyIA_lTD_VYk&redir_esc=y#v=onepage&q=proxmox&f=false.
- [11] Jordan MacPherson. “What Is a Hypervisor? – Types, Benefits How Does It Work?” In: (2022). URL: <https://www.parkplacetechnologies.com/blog/what-is-hypervisor-types-benefits/>.
- [12] Stefan Mischook. “Cloud Hosting vs Traditional Hosting”. In: (2016). URL: <https://www.youtube.com/watch?v=4lLskYgpKKo>.
- [13] Jonas Ohlin. “Transportstyrelsen – detta har hänt”. In: (2019). URL: <https://www.svt.se/nyheter/inrikes/transportstyrelsen-detta-har-hant>.
- [14] Pierangela Samarati, Indrajit Ray, and Indrakshi Ray. “Virtualization Technologies and Cloud Security: Advantages, Issues, and Perspectives”. eng. In: *From Database to Cyber Security*. Vol. 11170. Switzerland: Springer International Publishing AG, 2018, pp. 125–127. ISBN: 3030048330.
- [15] Synergy Research Group Statista. “Amazon, Microsoft Google Dominate Cloud Market”. In: (2022). URL: <https://www.statista.com/chart/18819/worldwide-market-share-of-leading-cloud-infrastructure-service-providers/>.