Research document infrastructure - Suit IT

PROJECT - SUITIT

BIES, KEANU K.G.A.J.

Table of contents

Introduction

In this semester we've got a project from a real IT Company, that is something that really gave me a motivational boost, because you are dealing with a real case that could be used in production.

Wen we've got the presentation about our project, I already was doing some brainstorming work, about all the possibilities.

The main challenge was to create a complete working environment without relying on Microsoft or any Microsoft licensing

This assignment is based on the case *"escaping from Microsoft"*. The company in this case is currently 100% dependent on Microsoft and is frustrated with the rising prices. They are looking for an open-source alternative that still offers the same level of usability and functionality. The goal is to design a fully open-source, user friendly workplace environment for a company with 50 to 100 employees, including operating system, office applications, IAM, Hybrid working.

What makes this case interesting is that other companies want more control, lower costs and less Microsoft in their organisation. Finding the balance between open source, security and user friendly is a big part of the challenge.

From the very beginning I started thinking about how different open-source solutions could be combined into one seamless environment, for example Linux Workstations, Office tools, I think this project doesn't only push us to just install software. But also give us an idea to design an ecosystem that feels the same for end users.

What is my part in this project

Durning this project, my personal focus within our team of six is the hybrid infrastructure. This means researching and designing the technical foundation that allows the environment to run partly on-premises and partly in the cloud. Completely without Microsoft technologies.

Central Question

The central question for my part of the project:

- *"How can I design a stable, scalable and maintainable opensource hybrid infrastructure for 50-100 employees."*

Sub questions

To really know what my main goal is, I need to define the central question into multiple sub questions.

1. Which open-source hypervisor is most suitable for hosting on premises servers?
2. How can an on-premises infrastructure components connect with cloud components in a hybrid setup?
3. Which storage technologies support hybrid usage with both local and cloud sync?
4. What network architecture is required for a secure hybrid environment?
5. What are the advantages and disadvantages of hybrid compared to a fully cloud based infrastructure?

Research

To answer my central question, I Researched the core components required to build a stable, scalable and maintainable open-source hybrid infrastructure. This research focuses on four infrastructure areas:

1. Virtualization
2. Hybrid cloud integration
3. Storage
4. Networking

Virtualization - Research

Viruialization is the foundation of any hybrid infrastructure because it allows many different servers and services to run on shared hardware. I compared three open-source hypervisors.

Proxmox VE - [Documentation](#)

Proxmox VE is a virtualization platform built on Debian that combines virtual machines and containers in one management interface. It is designed to provide a central place to manage servers, storage and backups in a on-premises environment.

Key Points:

- Web Based management interface
- KVM Virtual machine support
- Snapshots and backups
- Cluster functionality
- No licensing cost
- Community based documentation

XCP-ng – [Documentation](#)

XCP-ng is a virtualization platform based on the Xen hypervisor. It focuses on performance and stability and is typically managed through Xen orchestra, which provides centralised VM administration.

Key Points:

- Xen-based hypervisor
- Snapshot support
- Live VM Migration
- Multiple storage backends
- Optional Xen Orchestra management
- Fully open source

KVM/QEMU - [Documentation](#)

KVM is a virtualization component built directly into the Linux kernel. Combined with QEMU for hardware emulation. It provides full virtualization capabilities but requires separate tools for management.

Key Points:

- Integrated into the Linux kernel
- Hardware assisted virtualization
- Full OS virtualization
- Requires external management tools
- Open source

Virtualization – Conclusion

Based on the option I've searched for, Proxmox VE is the most suitable choice for the hybrid infrastructure. It provides a platform that supports both virtual machines and containers, includes snapshot and backup features and offers a fully functional web interface without licensing costs.

Why Proxmox VE

- Provides VMs and container in one platform
- Includes snapshot and backup features
- Web interface for management
- Fully functional without the use of a paid license

Why not XCP-ng

- Requires Xen Orchestra for full management, creates more complexity
- No support for native container support
- No full storage and backup features integrated.

Why not KVM/QEMU

- No build in web interface, need to use external application
- A lot of manual configurations required

Hybrid cloud integration – Research

Hybrid cloud integration focuses on connecting the on-premises infrastructure with AWS to maintain flexibility, scalability and reliability. The on-premises environment runs the core services (virtualizations, storage, networking), while AWS provides additional hybrid capabilities such as, the secondary DC, backups and optional cloud services. Think about PaaS or SaaS services. The goal is to keep the solution fully open source on premises while make use of other services where this adds value, without relying on Microsoft.

AWS S3 (object storage) – **Documentation**

AWS S3 could be used to store external objects for backups or long-term data storage and other hybrid storage situations

Key Points:
- Highly durable object storage
- Lifecycle policies for cost control
- Clients and backup tools
- Integrates with hybrid sync solutions

AWS EC2 – **Documentation**

EC2 could be used for extending compute capacity or hosting workloads that must be reachable externally.

Key Points:

- Automatic scalable
- Multiple OS images available
- VPN integration possible

Terraform – **Documentation**

Terraform is used to deploy and manage my infrastructure. From a single configuration file.

Key Points:
- Version-controlled deployments
- Reusable modules
- Declarative configuration
- Consistent provisioning

Hybrid cloud integration – Conclusion

Terraform, S3 and EC2 form a flexible hybrid architecture. The on-premises environment remains fully open source while AWS provides scalable storage and compute capability when needed.

Why Terraform + AWS S3/EC2

- Terraform manages the resources
- S3 provides off-site backup and object storage
- EC2 offers optional extra compute service's
- No Microsoft technologies involved.

Why no other cloud tooling:
- Azure is excluded due to the Microsoft requirement
- Google Cloud offers less integration with open-source hypervisors
- Many hybrid tools require commercial licenses

Storage - Research

Storage is essential for user data, shared folders, VM disks.

TrueNAS – Documentation

TrueNAS is a ZFS-based storage platform designed for reliability and flexible data management

Key Points:
- ZFS storage with snapshots
- SMB and NFS support
- Web Based management interface
- Cloud Sync

SAMBA – Documentation

 Samba provides SMB file sharing for Linux and Windows environments.

Key Points:

- SMB File sharing
- User and group permissions

NFS – Documentation

NFS is a lightweight file sharing protocol for Linux systems.

Key Points:

- Lightweight protocol
- Native for Linux.

Storage – Conclusion

TrueNAS is the most suitable storage platform for this hybrid infrastructure. ZFS offers snapshots and strong data integrity.

Why TrueNAS

- ZFS for data integrity
- Built-in snapshots
- SMB and NFS support
- Cloud Sync
- Full management interface

Why not samba Alone

- SMB only
- No snapshots or integrity features
- Depends on underlying filesystem

Why not NFS alone

- Lightweight but limited
- No advanced storage features

Networking – Research

Networking ensures proper segmentation, routing, secure access and hybrid connectivity. The network layer connects on premises systems with AWS through a secure VPN tunnel.

pfSense – [Documentation](#)

pfSense is an open-source firewall and routing platform used in enterprise environments.

Key Points:

- Firewall and routing
- VLAN support
- DHCP/DNS services
- VPN capabilities
- Web management interface

WireGuard – [Documentation](#)

WireGuard is used to create a secure, tunnel between the on-premises network and AWS.

Key Points:

- High-performance VPN
- Simple configuration

NGINX (Reverse Proxy - [Documentation](#)

Nginx is used to publish internal applications securely

Key Points:

- Reverse proxy
- Load balancing
- SSL/TLS
- Routing to internal services

Networking – Conclusion

pfSense, WireGuard and NGINX form a complete networking foundation for routing, segmentation, secure hybrid connectivity and controlled exposure of internal applications.

Why pfSense + WireGuard + NGINX

- pfSense handles firewalling, VLAN segmentation and routing
- Wire Guard provides fast and secure AWS and on premises connectivity
- NGINX enables secure access to internal applications
- Full documentation available

Why not OPNsense or OpenVPN

- PfSense has a broader documentation.
- OpenVPN is a more complex configuration

Conclusion

The researched components together form a complete open-source hybrid infrastructure that meets the requirements of the case. Proxmox VE provides virtualization, Terraform deploys my infrastructure, TrueNAS handles storage, and pfSense, Wire Guard and NGINX handle networking and hybrid connectivity.

This combination of services is stable, scalable and cost efficient to avoid Microsoft technologies.

Sources

Amazon Web Services. (2024). *Amazon EC2 documentation*.
https://docs.aws.amazon.com/ec2/

Amazon Web Services. (2024). *Amazon S3 documentation*.
https://docs.aws.amazon.com/s3/

Donenfeld, J. (2024). *WireGuard documentation*.
https://www.wireguard.com/

F5 NGINX. (2024). *NGINX documentation*.
https://docs.nginx.com/

HashiCorp. (2024). *Terraform documentation*.
https://developer.hashicorp.com/terraform/docs

iXsystems. (2024). *TrueNAS SCALE documentation*.
https://www.truenas.com/docs/scale/

Netgate. (2024). *pfSense documentation*.
https://docs.netgate.com/pfsense/en/latest/

Proxmox Server Solutions. (2024). *Proxmox VE administration guide*.
https://pve.proxmox.com/pve-docs/

QEMU Project. (2024). *QEMU documentation*.
https://www.qemu.org/docs/

Samba Team. (2024). *Samba documentation*.
https://www.samba.org/samba/docs/

Vates. (2024). *XCP-ng documentation*.
https://xcp-ng.org/docs/

Vates. (2024). *Xen Orchestra documentation*.
https://xen-orchestra.com/docs/

OpenAI. (2025). *ChatGPT (GPT-5.1)*.
https://chat.openai.com/