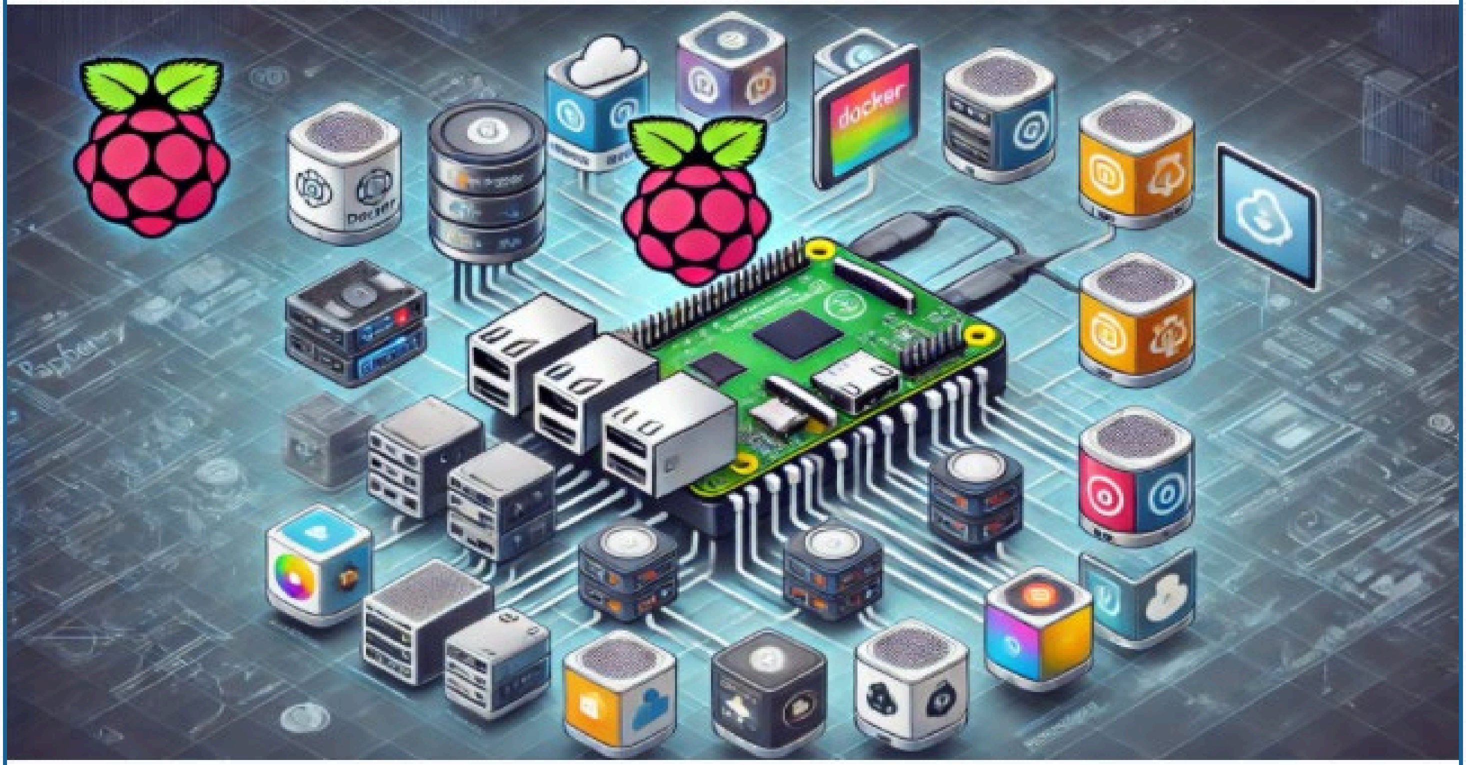
Lycée Guillaume Kroll

bts >> cloud computing

Esch/Alzette, the 18th December 2024

Project Documentation:

Privat Cloud Project



NGUYEN Auguste, MAHIC Refik, DURAES David

Table of Contents

1. Introduction

1.1 Project Objective

The goal of this project is to design and deploy a secure private cloud storage solution using Raspberry Pi, Docker, NextCloud and MariaDB. This solution provides an affordable, scalable, and secure alternative to commercial cloud services.

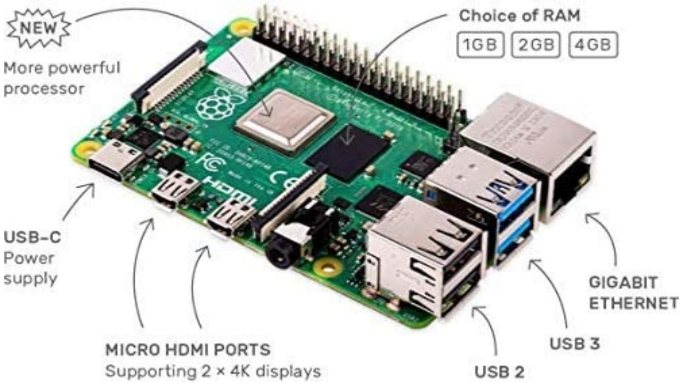
1.2 Scope

- Deploy NextCloud in a Docker container on Raspberry Pi 4B.
- Enable secure file storage, management, and access for multiple users.
- Provide a user-friendly interface and comprehensive documentation for installation, operation, and maintenance.

1.3 Deliverables

- Functional private cloud solution with NextCloud.
- Role-based user management and secure remote access.
- Documentation for architecture, administration, and agile practices.

2. System Architecture



2.1 Hardware Requirements

- Raspberry Pi 4B (4GB RAM).
- MicroSD card (16GB).
- Power supply and Ethernet cable.

2.2 Software Stack

- **Operating System:** Raspbian Lite 64-bit.
- **Containerization:** Docker and Docker Compose.
- **Application:** NextCloud.
- **Database:** MariaDB
- **Networking:** VPN or HTTPS secured with Let's Encrypt.

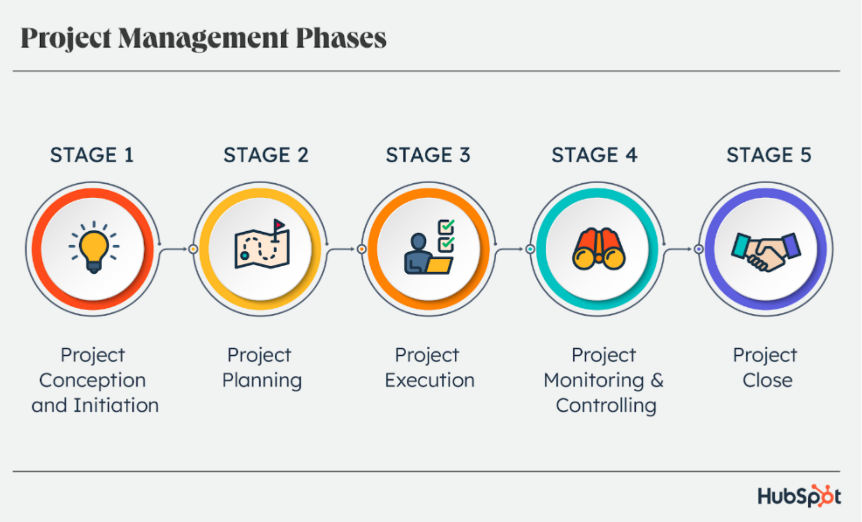
3. Planning Phase

3.1 Objectives:

- Plan deliverables and prioritize tasks.

3.2 Steps:

1. **Backlog Creation**:
    - Write detailed user stories (e.g., "As a user, I want to securely upload files so that my data remains private").
    - Prioritize tasks using the MoSCoW method.
2. **Sprint Planning**:
    - Decompose prioritized user stories into smaller, actionable tasks that can be completed within a single sprint.
    - Estimate the effort required for each task using techniques like story points or T-shirt sizing.
    - Assign tasks to team members based on their expertise and availability.
    - Define clear and measurable sprint goals, ensuring they align with the overall project objectives.
    - Create a sprint backlog in tools like Planner


Project Management Phases
STAGE 1
Project Conception and Initiation
STAGE 2
Project Planning
STAGE 3
Project Execution
STAGE 4
Project Monitoring & Controlling
STAGE 5
Project Close
HubSpot

4. Execution Phase

4.1 Setup

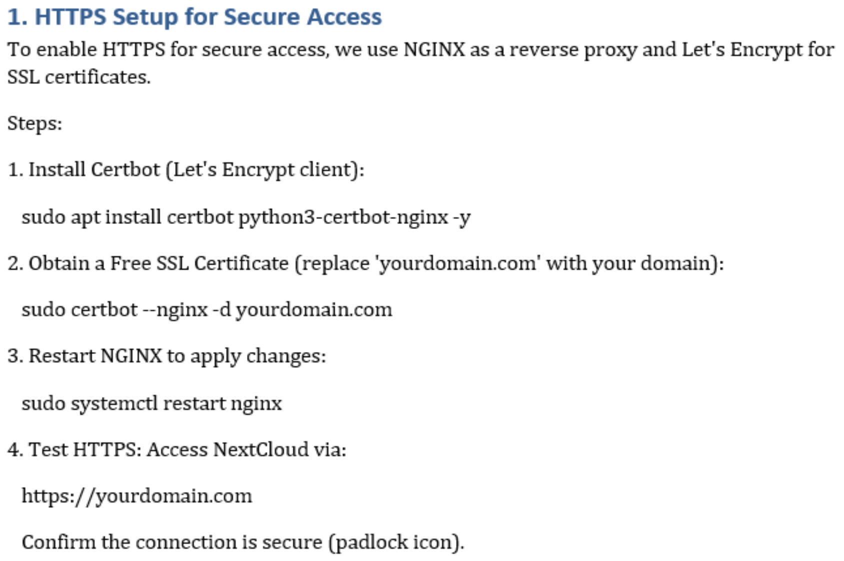
1. Install Raspbian Lite (64-bit) on Raspberry Pi Model 4B.
   Update the system and install Docker.
2. Configure Docker Compose for NextCloud.
3. Configure the Raspberry Pi for stable network access, including enabling and setting up SSH for remote management.
4. Set up persistent storage volumes for data and database.

4.2 NextCloud Configuration

1. Deploy the NextCloud container.
2. Initialize the NextCloud instance and create the admin user.
3. Configure user roles and permissions for multi-user access.

4.3 Security Configuration

1. Set up SSL/TLS encryption using Let's Encrypt.
2. Configure secure remote access via VPN or HTTPS.



1. HTTPS Setup for Secure Access

To enable HTTPS for secure access, we use NGINX as a reverse proxy and Let's Encrypt for SSL certificates.

Steps:

1. Install Certbot (Let's Encrypt client):

```
sudo apt install certbot python3-certbot-nginx -y
```

2. Obtain a Free SSL Certificate (replace 'yourdomain.com' with your domain):

```
sudo certbot --nginx -d yourdomain.com
```

3. Restart NGINX to apply changes:

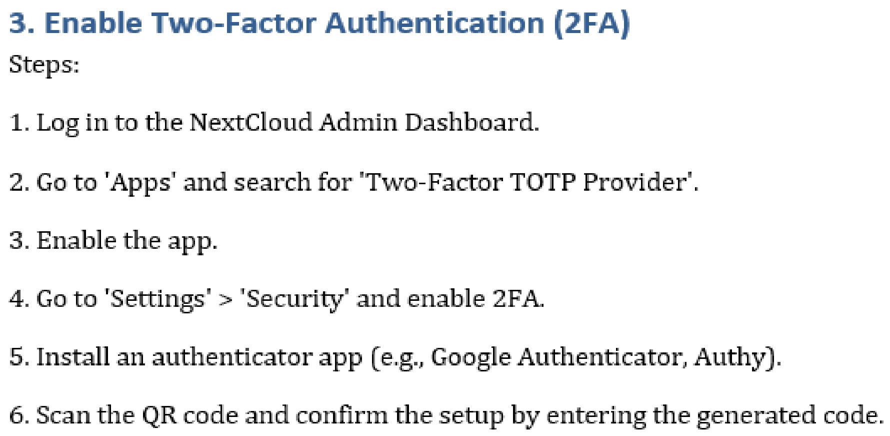
```
sudo systemctl restart nginx
```

4. Test HTTPS: Access NextCloud via:

https://yourdomain.com

Confirm the connection is secure (padlock icon).

3. Implement Two-Factor Authentication.



3. Enable Two-Factor Authentication (2FA)

Steps:

1. Log in to the NextCloud Admin Dashboard.

2. Go to 'Apps' and search for 'Two-Factor TOTP Provider'.

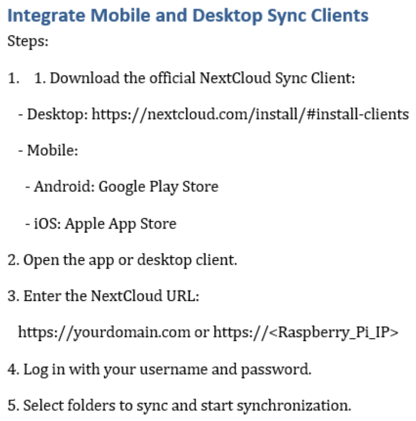
3. Enable the app.

4. Go to 'Settings' > 'Security' and enable 2FA.

5. Install an authenticator app (e.g., Google Authenticator, Authy).

6. Scan the QR code and confirm the setup by entering the generated code.

4.4 Testing



Integrate Mobile and Desktop Sync Clients

Steps:

1. 1. Download the official NextCloud Sync Client:

  - Desktop: https://nextcloud.com/install/#install-clients

  - Mobile:

    - Android: Google Play Store

    - iOS: Apple App Store

2. Open the app or desktop client.

3. Enter the NextCloud URL:

  https://yourdomain.com or https://<Raspberry_Pi_IP>

4. Log in with your username and password.

5. Select folders to sync and start synchronization.

1. Update the file which contains all passwords
2. Test file upload, download, and organization features.
3. Simulate multiple concurrent users to evaluate performance.
4. Verify encryption and secure remote access.

5. Agile Practices

5.1 Framework

The project follows the Scrum methodology, which organizes work into weekly sprints. Each sprint includes regular activities such as stand-up meetings for progress updates, sprint planning sessions to define tasks, and retrospectives to assess what went well and what can be improved.

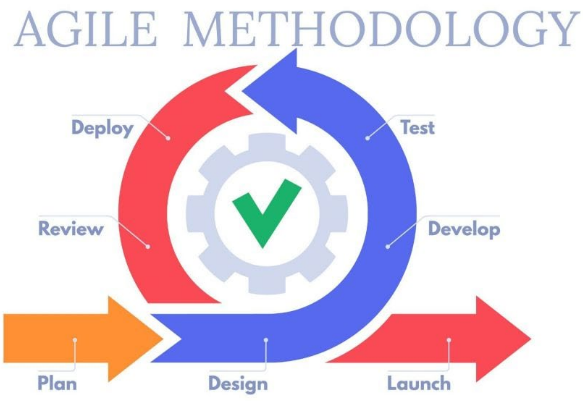
5.2 Backlog Management

User stories are prioritized using the MoSCoW method, ensuring that "Must Have" tasks are completed first. Tasks are organized into sprints and tracked using tools like Planner providing a clear view of progress and priorities.

5.3 Benefits of Agile Practices

The use of Agile practices offers several advantages:

- Incremental delivery for early feedback.
- Flexibility to adapt to changing requirements.
- Continuous improvement through retrospectives.



6. Administration Documentation

6.1 Routine Tasks

- **Backups**: Develop a backup strategy that includes daily automated backups of NextCloud data and the MariaDB database. Use tools like rsync or mysqldump for backups, and ensure backups are stored on a separate storage medium or cloud storage for redundancy.
- **Updates**: Regularly check for updates to Docker images, the NextCloud platform, and associated plugins. Implement a version control strategy to test updates in a staging environment before deploying them to production.
- **Monitoring**: Use monitoring tools such as htop for CPU and memory usage, iotop for disk I/O, and netstat for network activity. Set up alert systems to notify the team of performance issues or resource exhaustion.

6.2 Troubleshooting

- Restart services if containers crash.
- Check logs for errors and unusual activities.
- Restore data from backups when necessary.

6.3 User Management

- Create, delete, and manage user accounts.
- Assign roles and permissions based on user needs.

7. Conclusion

The documented process provides a comprehensive guide to planning, executing, and managing the Privat Cloud Project. By adhering to these processes, the team ensures successful delivery of a secure, scalable, and user-friendly private cloud solution while fostering skill development and collaboration

8. Appendix

8.1 References

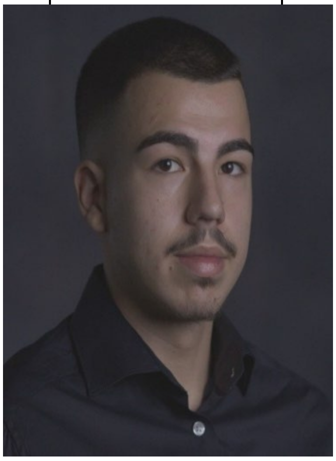
- NextCloud Documentation: https://docs.nextcloud.com/
- Docker Documentation: https://docs.docker.com/
- Raspberry Pi Resources: https://www.raspberrypi.org/resources/

8.2 Acknowledgments

Special thanks to all team members for their contributions and collaboration throughout the project lifecycle.

MAHIC Refik

DURAES David



NGUYEN Auguste