POLITECNICO DI TORINO

DEPARTMENT OF CONTROL AND COMPUTER ENGINEERING (DAUIN)

Master Degree in Computer Engineering

Master Degree Thesis

Custom cloud storage solutions based on Nextcloud: a case study implementation

Author: Massimo Missio

Advisor: Prof. Paolo Ernesto Prinetto
Co-Advisor(s): Dr. Matteo Fornero, Dr. Nicoló Maunero

September, 2022

Abstract

Nowadays, the world is characterized by a rising and relentless production and consumption of information, be it related to the private life of people or to aspects concerning businesses, industries, organizations, and Nations.
Globalization demands information to be available always and anywhere. The constant availability of information is at the very core of a concept known as business-continuity which, however, can also be applied to the private dimension of our lives, since today most of the information we make use of is not really under our control, rather, it is stored somewhere around the world.
Making data always available for people, independently from their geographic location, from the device they are using, and from many other parameters, is not a simple task. Most companies and organizations do not have the budget and the technical knowledge to fulfill such a task on their own, which is why they buy services from providers offering data storage solutions based on the 'Cloud Paradigm'.
Cloud storage services provided by vendors such as Dropbox, Google, Amazon, Microsoft, Apple, etc. are extremely popular among private users, companies, and organizations; however, they all have common problems: data security and data sovereignty.
It is not always possible, in fact, to precisely determine where data are physically stored (i.e., even if the provider is GDPR-compliant); moreover, information about the cybersecurity policies and procedures adopted by providers are not usually very detailed. These issues concern anyone, but they are clearly more important for actors dealing with confidential data which might have an impact on the safety and security of people and other valuable assets. There exists, therefore, a space for designing and implementing cloud storage solutions that give customers and users a greater control over their data. The aim of this thesis is to investigate the design and development of cloud storage solutions capable of giving users a complete control of the data and of the underlying infrastructures including, for instance, how data are managed and protected.
With respect to this goal, open-source cloud storage platforms represent an ideal starting point being inexpensive, highly customizable, and open to anybody's contribution for improvement. The open-source cloud storage platform chosen for this thesis is Nextcloud, which is probably the most popular solution in its category.
Starting from an implementation based on a very simple infrastructure, this thesis investigates the deployment of Nextcloud according to multiple strategies, such as purely on-premise, purely remote, and hybrid ones.
Issues such as confidentiality, integrity, authentication, authorization, and physical control

of data have been taken into account during the analysis, also considering the native features provided by Nextcloud.
In conclusion, this thesis discusses the possibility of implementing a secure, highly-customized cloud storage platform based on open-source software that is capable of competing with, and even surpassing, commercial solutions available on the market.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In his book "The Road Ahead" (1995)[11], Bill Gates wrote that he believed there would come a day when people could do business, study, learn about the world and its cultures, attend important shows, meet new friends, buy products, show photographs to distant relatives, all without moving from their desks or armchairs. That day, in fact, has already come a long time ago.
Nowadays, very few people could live without making use of modern technology, especially smartphones, PCs, and Internet. Information Technology is so important that Google, for instance, has become an integral part of our everyday lives: the name itself has become a verb included in the dictionary (i.e., to google something). IT has progressively assumed an essential role in modern society bringing, as always, advantages and disadvantages. At first glance, IT technologies simplify many aspects of people's life; however, they also imply a somehow hidden complexity that is not easy to comprehend for most people. With respect to the past, we have many more possibilities: we can work and study from home, we can buy stuff online from the other side of the world and get it delivered at our door in a few days, we can take thousands of pictures with our smartphone and do not worry about loosing them because we synchronize them to our cloud account. Additionally, this trend has been sped up by the Covid-19 pandemic which forced millions of people to work from home for months.
One of the common traits of this evolution in our society is represented by the constant necessity of being able to access to data, anywhere and anytime. This is especially important when considering data concerning businesses, industries, and organizations. Remote working would not be possible if data were not constantly available to all employees working according to different schedules and geographic locations, meaning that data availability is essential to grant business continuity.
Making data always available to anybody or, depending on their nature, to selected people, is not a simple task. It requires a deep understanding of several issues, such as reliability and security of the system where data are stored and managed. Because of the complexity of this task, not only individuals, but also companies and organizations make use of services specifically provided for this purpose. Nowadays, everyone knows what Google Drive, Dropbox, iCloud, etc. are. These products belong to a market sector, usually known as 'cloud storage', that has grown significantly in the last decade. In more general terms, cloud storage belongs to a paradigm that is known as 'cloud computing'. Cloud computing arose

because of many factors, mainly the widespread availability of fast Internet connections and the need to centralize data, both for individual users and for companies/organizations. Keeping all data in the same place, managed by a third party, without having to worry about how to store them, how to protect them, how to guarantee high availability and reliability, is a major advantage of cloud storage solutions.
Clearly, cloud storage also brings disadvantages and problems. While cost might be the most obvious problem for most people and companies, there are other issues that are far more important. Data security, for instance, is always a major concern when considering cloud storage services. Specifically, we are talking about giving a third party (the cloud storage provider) control over data confidentiality, physical data management, data access, information sharing policies, and much more. These issues must be carefully evaluated, especially when dealing with sensitive data whose unauthorized access/usage might have a negative impact on multiple aspects (i.e., people's safety, business continuity, etc.).
Cloud storage providers (i.e., Dropbox, Google, Amazon, etc.) usually agree on periodic security assessment activities conducted by independent agencies to verify and review the compliance of the cloud storage infrastructure to the highest cybersecurity standards (or, at least, to the minimum levels agreed with the customers). Providers usually assert that they protect data in motion, meaning data that is being transmitted from a place to another (i.e., from a server to the PC of the user), using state-of-the-art encryption protocols such as TLS and SSL [1]. Moreover, data at rest [2] are usually protected by means of strong encryption algorithms such AES-256[3]. Sometimes, providers also state that they cannot break data confidentiality in any way because their systems rely on state-of-the-art protocols and techniques such as ZKP [4]. However, the user/customer cannot verify any of these claims since the cloud storage services of traditional providers are commercial, meaning that they are closed systems which do not give external people (i.e., everyone except those who develop and maintain the cloud storage service) any insights about how they are actually implemented (i.e., the source code is not available for review).
Given the problems that have been described, there exists a space for exploring solutions that can bring to the users the benefits of a cloud storage service without experiencing its typical disadvantages. The goal is, therefore, to combine the advantages of cloud storage with the advantages of an on-premise, custom data storage infrastructure. The aim of this thesis, in fact, is to investigate the design and development of cloud storage solutions capable of giving customers full control over the data and the underlying infrastructures including, for instance, how data are managed and protected. With respect to this goal, open-source cloud storage platforms represent an ideal starting point being inexpensive, highly customizable, and open to anybody's contribution for improvement. Moreover, being open-source, their source code can be thoroughly analyzed searching for backdoors, bugs, vulnerabilities, and so on. This is an enormous advantage for companies and organizations that are willing to put some effort in improving the security of the platform.

[1] Transport Layer Security and Secure Sockets Layer, for more details see [6]

[2] This term refers to data stored in local and remote storage units.

[3] Advanced Encryption Standard [47]

[4] Zero Knowledge Protocol [3]

Finally, it must be noted that, since the source code is publicly available, the entire community of developers and cybersecurity expert might be interested in contributing to the development and improvement of such a cloud storage solution.
The open-source cloud storage platform chosen for this thesis is Nextcloud [27], a fully open-source cloud solution that was created started in 2016 as a fork of OwnCloud[41]. On its website, Nextcloud is defined as: *a technology that combines the convenience and ease of use of consumer-grade solutions like Dropbox and Google Drive with the security, privacy and control that business needs*[19]. Indeed, Nextcloud is charachterized by a user-friendly platform that is very similar to those offered by commercial competitors, but it also provides significant customization possibilities (being open-source) and full control over privacy and security issues.
This thesis has been structured around the design of some cloud-storage prototypes based on Nextcloud. The first prototype is focused on a fully on-premise infrastructure where the company or private individual manages the entire system, from hardware installation/-maintenance to the higher abstraction layers (i.e. user management, user permissions, etc.). The other prototypes adopt a more complex and realistic approach where data are no longer stored on a private infrastructure that is deployed on-premise, on the contrary, they are stored on a remote infrastructure provided by third parties (i.e. Google, Amazon, etc.) whose only role is to supply the hardware needed to deploy the cloud platform. The third party is also responsible for addressing hardware-related issues that might impact on reliability and availability of the system, such as power supply and hard drive failures. In this case, it must be noted that data security is not a concern of the third party, meaning that the owner of the cloud platform still keeps control of how data are encrypted (i.e., they are sent to the third party already encrypted). Finally, this thesis discusses advantages and disadvantages of the proposed prototypes, analyzing risks and benefits to determine whether it might make sense to deploy a ‘private‘ cloud infrastructure.

The remainder of this document is organized as follows. In Chapter 2, "State of the art", the Cloud paradigm will be defined, the most used commercial cloud services and also some small-case, on-premise alternatives to cloud storage (NAS systems) will be analyzed; in Chapter 3, "Background", Nextcloud and its main characteristics and features will be explored; moreover, NFS, SMB and SSHFS and some VPN solutions will be defined; in Chapter 4, "Implementation", different Nextcloud-based cloud prototypes will be studied and analyzed; in Chapter 5, "Results", achievements will be shown; in Chapter 6, "Conclusion", a brief recap of the thesis will be reported along with final thoughts and considerations.

Chapter 2

Background

2.1 Cloud Computing

Cloud computing is a technology that offers a set of network-hosted services; thanks to cloud computing, it is possible to obtain resources that can be easily configured according to one's needs and accessed directly from the Internet. Different companies often have different needs and, as a result, opt for different types of cloud. There are mainly 3 cloud computing models that differ, essentially, in the way the service is delivered: public cloud, private cloud, hybrid cloud.

- **Public cloud**: this refers to the provision of services that are based on an IT infrastructure belonging to the service provider. This type of cloud computing involves the transfer of all company data from its own servers to those of the provider. Therefore, it allows the company not to have to bear the costs of maintaining and managing the infrastructure, while still being able to take advantage, via the Internet, of fundamental services such as storage capacity, computing power, etc.

- **Private cloud**: the IT infrastructure is owned by the individual company, which has exclusive control over data and resources. It is advisable to adopt this solution when the company has specialized personnel capable of managing the technological infrastructure. A private cloud may be physically located within the company's local data center, in which case it is referred to as an on-premise solution. The cloud may also be hosted by a third-party provider, but even in the latter case, access is for the exclusive use of the company that owns it. It should be noted that in the case of an on-premise solution, the machines, including servers and associated components, are running within the boundaries of the company. Since the cloud infrastructure is managed on-premises, it is the company itself that has to manage the hardware and is also responsible for its maintenance and all related processes. This gives the company greater control and security and, in addition, the possibility to pay only for the resources that are used and to offer enormous flexibility. In the case of a private cloud hosted by the provider, on the other hand, the provider will be responsible for maintaining the hardware and software updates. The customer, on the other hand, will have control over server resources such as CPU cores, RAM and storage space.

- **Hybrid cloud**: this is an intermediate solution between public and private clouds. There will therefore be services based on one's own corporate infrastructure and others provided by the cloud provider.

2.1.1 Main categories of cloud computing

- **SaaS**: Software as a service is a cloud computing model in which software is hosted by third-party providers and made available through the Internet. It is a type of cloud that is becoming increasingly popular, as it allows customers not to worry about software updates, compatibility, accessibility and so on; in fact, all these operations are handled directly by the service provider.
- **PaaS**: Platform as a service is a cloud computing model that offers users applications on top of an infrastructure. In particular, a PaaS provider provides the tools for application development. This services are generally hosted on hardware and software belonging to the provider's infrastructure, exempting end users from server maintenance. In general, companies use PaaS not to replace the entire corporate infrastructure, but only to take advantage of the service that allows applications to be developed without having to purchase and install additional hardware on site.
- **IaaS**: Infrastructure as a service falls into the category of cloud computing, where a third-party provider hosts virtualized computing resources over the Internet. In addition to providing hardware, software, servers, storage, etc., IaaS providers also host users' applications and help them manage system maintenance, including backup operations and resilience planning. The resources offered are highly scalable and can be adapted on demand.

2.1.2 VPS and Dedicated Server

VPS (Virtual Private Server) and Dedicated Server are two different types of solutions for renting from an external provider, a server: in the first case, a virtual one, in the second, a physical one. The VPS falls into the category of IaaS (Infrastructure as a Service) cloud computing, whereas the dedicated Server could be cataloged as an evolution of the same category, in that, the client does not have control of a virtual machine installed on top of the server, but, has direct control of the physical server. It has been decided to go into these two solutions specifically, since they will both be considered in 3.

- **VPS**: a Virtual Private Server is a virtual machine running on a physical server and, therefore, offers users all the typical functionalities that a dedicated server would provide. Providers offering hosting services typically host more than one VPS on a single physical server, and each of these has a dedicated operating system (OS) to which full root access is provided. This feature allows each system administrator to work autonomously, despite the hardware base being shared with other users. The hypervisor [1], the software component that determines the virtualization of the environment, is

[1]The hypervisor allows several virtual machines to run on a single server and at the same time ensures that they do not interfere with each other and that they all have the necessary capabilities [43].

Table 2.1. Virtual Private Server vs Dedicated Server

Characteristics	VPS	Dedicated Server
Price	Expensive, but cheaper than the dedicated server: from 10 dollars to 150 dollars	The most expensive solution: from 100 dollars upwards
Performance	High performance defined by the physical resources available according to the plan, possible slowdowns if several VPSs on the server have high traffic simultaneously.	The highest performance available according to the actual hardware of the machine.
Security	Partially vulnerable if other VPSs on the server are subject of a security breach	Maximum security as there is exclusive access to the server and the possibility of installing specific software and firewalls
Scalability	Ability to scale up by simply requesting more resources from the provider	It already has maximum performance, it is only allowed to scale vertically requiring better hardware if possible

used to manage the hardware; in fact, it divides the physical resources, such as CPU, RAM or hard disk space, between the various VPSs. Within a VPS the administrator user, using root permissions, is allowed to install any applications such as a web server, mail server, etc., on the previously configured operating system. A VPS is an excellent solution that offers flexibility at a high, but not excessive cost (more details in table 2.1). However, from a security point of view, you are still partially vulnerable if other VPSs on the same server suffer a security breach. Furthermore, even if the hypervisor provides a fixed amount of resources, the performance of the system may be affected by the other VPSs located on the server. Indeed, often the number of network cards and LAN connections is less than the number of VPSs provided by the server, which means that it will be necessary to share the network card with other customers who have purchased a VPS.

- **Dedicated Server**: is a server, located in a Server Farm [45] or datacentre, dedicated entirely to a specific user. In this case, the provider only takes care of hardware and connectivity issues at the request of the customers themselves. The customer is thus provided with a physical machine in its entirety, on which he can work remotely; this type of solution delegates the customer from the duty of having to purchase and install the hardware on his own premises and brings with it significant savings related both to the actual purchase of the machine, but also to the management and maintenance of the physical server itself (such as the costs of powering the machine). In a dedicated server, the machine is not divided among several users but is entirely dedicated to a single client; this makes the environment more stable and perfect for processing large amounts of data. Furthermore, by having direct access to the server, one has the

possibility of fully exploiting the underlying hardware architecture, which leads to a significant improvement in performance. This type of service offers a higher level of security since you have full control of the machine; however, in addition to having a higher cost, it obviously requires a trained technical staff capable of managing the server.

Table 2.1 shows a comparison between VPS and dedicated server.

2.2 Virtual Private Network (VPN)

A VPN (Virtual Private Network) is a hardware and/or software technology with which a secure and reliable communication service is created between two machines when they are in a public or untrusted network. With a VPN, confidentiality, integrity and authentication can be achieved. It is called 'virtual' because the two machines communicate as if they belonged to the same private network, but, in reality, this is not the case: this technology exploits the public structure, i.e. the Internet, to create this 'logical' communication. A VPN network therefore establishes a private connection between the two machines located in different physical locations by encapsulating the traffic within a virtual 'tunnel' that passes through the public nodes of the Internet.

IPsec (Internet protocol Security) is a protocol that provides level 3 security for IPv4 (also for IPv6) and can be used to create Virtual Private Networks (VPNs). At the network level, it provides secure communication through an IP packet authentication and encryption mechanism. IPsec-based VPNs are mainly divided into two modes: transport-mode and tunnel-mode.

2.2.1 Transport-mode (end to end security)

This method consists of enabling IPsec directly on the end nodes of the communication. A 'secure virtual channel' is thus established directly between the two hosts. It is necessary for IPsec to be supported and installed on the two hosts, which must necessarily have the appropriate cryptographic capabilities. This type of mode is independent of everything: it is therefore not necessary to worry if the LAN (Local Area Network) is not secure, if the gateway is managed by untrusted people, or even the WAN (Wide Area Network), as communication is protected from the end nodes. This is a light and permissive solution, but it only guarantees the protection and encryption of the packet payload, in fact, the IPv4 header will remain unencrypted. A diagram of such a solution can be seen in 2.1.

2.2.2 Tunnel-mode (Basic VPN)

This mode, also called Basic VPN, creates a 'secure virtual channel' between the two gateways of the communication; as can be seen in the figure 2.2, it is therefore sufficient for IPsec to be installed only on the gateways. Consequently, it is not necessary for the two end nodes of the communication to have cryptographic capabilities to support IPsec. Through this method, the entire original packet (both the payload and the IPv4 header) is encrypted and placed within another IPv4 packet; this results in an increase in packet size. This solution is computationally heavy; moreover, since the VPN starts from the gateways,

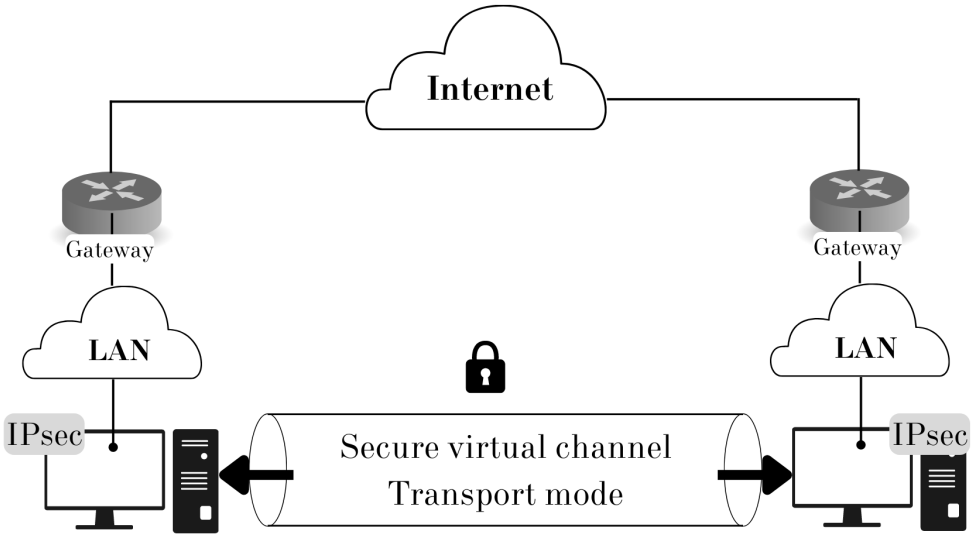

Figure 2.1. Schema of the *Transport mode IPsec*

communications from the host to the gateway are not protected: if LANs are not secure or if the gateways are managed by untrusted people, it is possible to be subject to attacks of any kind.

2.2.3 End to end security with Basic VPN

This solution is a combination of using IPsec in transport-mode and tunnel-mode. Two 'secure virtual channels' will therefore be created, one in transport-mode between the two end nodes, and the other in tunnel-mode between the two gateways. This creates a double line of defence: this solution is an example of the principle *defence in depth*[9]. As can be seen in the figure 2.3, IPsec will be installed both on the gateways and on the end hosts, which must have the necessary cryptographic capabilities to support it. With this type of solution, one will therefore have both the protection of the payload between the two nodes, and the protection of the entire original IPv4 packet between the two gateways. Obviously, this double line of defence makes this solution the most secure, but also the most expensive in terms of computation and system performance.

2.3 Nextcloud

It is a completely open-source file sharing and synchronization software that allows work teams to access and manage their data easily; and, thus, enables the creation of a personal cloud storage service. Today, it is widely used in different areas and offers a great solution,

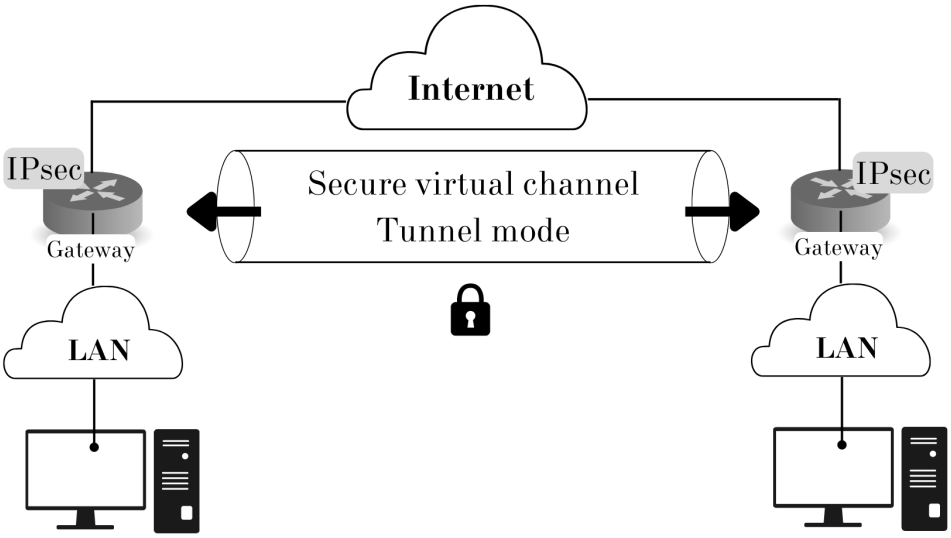

Figure 2.2. Schema of the *Tunnel mode IPsec*

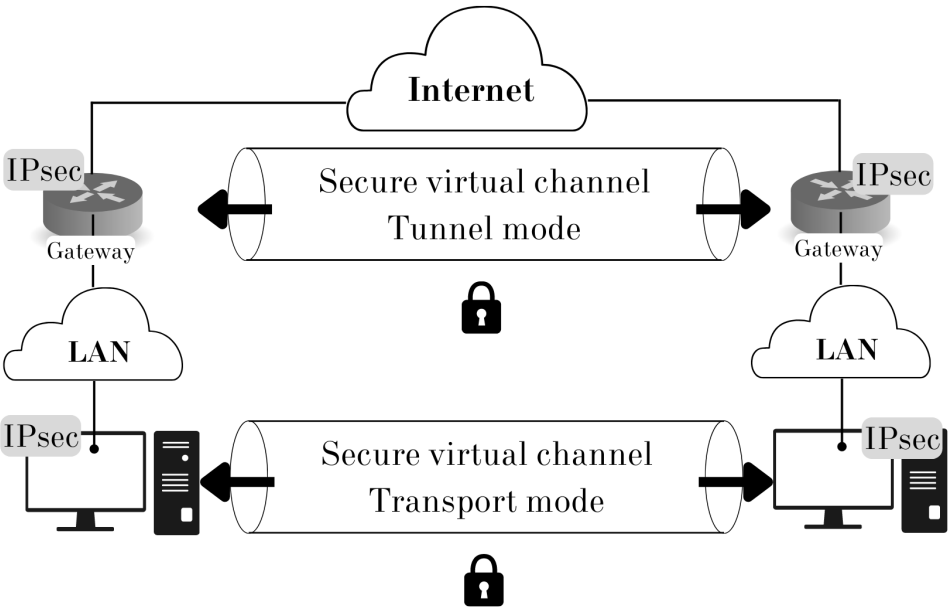

Figure 2.3. Schema of the *End to end security with Basic VPN*

for schools, businesses, and companies. It was born in 2016 as a fork of the OwnCloud project [41] developed by some of the same members of the original OwnCloud team. Nextcloud offers a large number of completely free features, ranging from the addition of communication methods via audio and video, to functions that aim to achieve better system security, such as protection from brute-force attacks, end-to-end encryption, etc. It is one of the most popular open-source projects in the world and is supported by a huge community. Thanks to its many functionalities and the possibility to be customized, the system allows one to create an 'ad hoc' cloud storage platform in which one has full control.

2.3.1 Nextcloud overview

Nextcloud comes in two versions: Nextcloud Server and Nextcloud Client. The Nextcloud Server represents the heart of the system; it is the software that must be installed on the machine that will provide the cloud storage service and from which, the entire system will be managed by the system administrator. Supported operating systems are: Ubuntu, Red Hat, Debian, SUSE and centOS (more details in the official docs [30]). The Nextcloud Client is, on the other hand, the software that is installed by various users to connect to the Nextcloud platform through their own devices. There is a Desktop version that supports Windows, macOS and Linux and also a Mobile version for both Android and iOS. There are therefore three methods of using Nextcloud: browser, Desktop Client and Mobile app.

- **Browser**: supported browsers are Mozilla Firefox, Google Chrome/Chromium, Apple Safari and Microsoft Edge. Users can use all the functionalities provided by Nextcloud through the browser: they can create, edit and delete files without necessarily having to download them (direct streaming is possible). Users also have the possibility of adding new folders and sharing files with other users by setting sharing rules, using all the apps provided by the system (previously installed and enabled by the administrator) such as reading and writing e-mails, or applications such as Online Office to work with online documents, setting notifications, etc.
- **Desktop Client**: as soon as it is installed and logged in, a folder is created on the user's device, called, Nextcloud. Inside this folder, the various files and folders of the user who is logged in will be found: there is the possibility to choose what to synchronize and what not to synchronize, thus choosing whether to create a local copy for each file and/or folder or to leave everything in the cloud. It is possible to add and edit files directly from this folder. The Desktop Client has, however, significant limitations in the sense that many functionalities are not available. For instance, when trying to open an e-mail notification, the browser will automatically open. The only available functionalities are:
  - add, edit or delete a file
  - consult notifications
  - enable CSE (Client Side Encryption)
  - synchronise files
- **Mobile app**: The "Nextcloud app" offers functions almost identical to those available with the Desktop Client: it is possible to create and edit files directly from the

application without necessarily having to download the content, and it is also possible to choose which files and folders to synchronise on the device in order to have a copy locally. Like the Desktop Client, however, this application also has limitations: some functions are only possible by connecting to the account via the browser. Again, the only functionalities available are:

  - add, edit or delete a file
  - consult notifications
  - enable CSE (Client Side Encryption)
  - synchronise files

  There is a further application, called "Nextcloud Talk", with which it is instead possible to communicate via audio or video call with Nextcloud users.

Users in Nextcloud can have different roles; the 3 main user types are: system administrator, group administrator, user.

- **System administrator**: the system administrator, most commonly called administrator, is the type of user who has control over the entire system. By connecting to the web interface with his or her access credentials, he or she has the possibility of installing, enabling or disabling applications, can configure other system functions (such as parameters for password policies), enable Server Side Encryption, Two Factor Authenticator, etc. In addition, it can create new users or delete them, and/or create new groups, for which it will be possible to specify the group administrator. The system administrator represents, therefore, the administrator of the entire Nextcloud platform and, has powers and privileges to install and modify any type of system functionality. Obviously, he then has all the standard user functionalities and has, therefore, the ability to create, edit and share files with other users.
- **Group Administrator**: is a user who has been placed in a specific group and assigned the role of group administrator: this role allows such a user to be able to add and/or delete users within the same group. The remaining capabilities are the same as for classic users.
- **User**: this is the standard user; it can only be created by an administrator or group administrator, it has a clearly defined amount of storage space, assigned during creation by the administrator. This user only has the possibility of adding, reading and editing files that he owns or files that have been shared with him or with the group to which he may belong. He can edit his personal information, change his password and set notifications. He cannot change any system functions and can only use the applications made available by the administrator.

2.3.2 Main features

In addition to simple file management, Nextcloud offers many additional functionalities: these are provided by the many different apps that allow, for instance, mail management directly from the web interface, online document editing and other functionalities such as

real-time audio and video communication methods. The main functionalities of Nextcloud are listed below.

- **Mail**: the mail app provides the administrator the ability to configure the system's mail server so that it can be used for sending notifications and resetting passwords. In addition, each user can link their personal mail to their Nextcloud account so that they can consult and send e-mails directly from the web interface [34].
- **Notifications**: notifications can be set up and changed according to the needs of each individual user. There are two types of notifications: push notifications and e-mail notifications. Push notifications are sent to the user on the browser, Dekstop Client or Mobile app, while e-mail notifications are received on the user's personal e-mail (only if this has been configured in advance). Notifications are fully customizable and can be received in these cases:
  - A file or folder is changed, shared or downloaded by mail/public link.
  - A calendar event is modified
  - A contact or address book is modified
  - Group membership are modified
  - Password or email are modified
  - For comments under file
  - Notifications related to system security
- **File sharing**: files and folders can be shared between users and groups. Permissions and privileges can be set for each file or folder, thus defining, specifically, the permissions to read, edit and delete files. Normally, files and folders can only be shared between users who have an account on Nextcloud, but there is the possibility of sharing files, via public links, with users outside the platform (this functionality can however be disabled by the administrator)[24]. For each file/folder there is the possibility of sharing it with users and/or groups. There is also the possibility of managing file access control, file storage and file labelling.
- **Online Office and PDF viewer**: these are two very powerful applications that offer users the possibility of simultaneously viewing and working with the main document formats (docs, exe, PDF, etc.) directly from the web interface without having to download these files locally [28].
- **External storage**: This application offers the possibility of mounting an external storage service or device as Nextcloud secondary storage. The administrator can also provide the possibility for each user to mount their own personal secondary storage. The services and devices that can be used as external storage are: Amazon S3, FTP, SFTP, SMB/CIFS, etc. [22]. When the administrator decides to add secondary storage of this type to the system, a specific directory (with the name determined by the admin) will be created for each user. This directory represents additional storage space that will, however, be shared equally between all users. Each user will be able to read, modify or delete files and folders within the directory, regardless of whether

or not they have been created by him previously. It, therefore, represents a shared space between all users where it is not possible to have one's own private space to manage personal files.

- **REDIS caching**: Nextcloud supports Redis caching, but it is not possible to install this feature directly from the web interface: the only way to enable it is to connect via SSH to the Nextcloud Server and install and configure Redis from the command line. Redis stores data in the RAM memory; if this data is requested by the system, it will be made available immediately and quickly, as it will not be necessary to query the database. The operating principle is identical to that of a cache memory for database queries, in fact, it provides an improvement in the system's read and write performance

- **Cron Job**: Nextcloud periodically needs to perform background jobs (i.e., databases cleanup). The default configuration is based on AJAX, but the recommendation of the official documentation is to use the Cron Job, which enables background jobs to be performed periodically.

- **Other functionalities**: Nextcloud offers many other applications that provide services of various kinds such as the possibility of communicating between users via audio or video chat, weather, music or radio applications. It is, therefore, a platform with a wide range of free applications available. It is possible to consult the official website for more details [20].

2.3.3 Security features

Nextcloud offers total control over personal data and it is to this that it owes its notoriety. In order to realise a secure infrastructure, it is of course necessary to put the correct protection measures in place. The main security features that can be used on Nextcloud are shown below.

- **2FA**: it is possible to enable directly from the web interface the Two Factor Authenticator through which there is a more secure authentication method that requires two forms of identification to access one's account. Several apps are available for 2FA, including TOTP [2] (which supports Google authenticator) or verification apps via SMS or Telegram, etc. (more details in the official doc [31]).

- **Password policies**: the administrator can set password policies directly from the browser, it is possible to set the minimum number of characters, force upper and lower case, use of special characters, etc. In addition, it is possible to specify the expiration date of passwords and to check the most commonly used passwords (more details in the official doc [32]).

- **Brute force protection**: Nextcloud offers native brute force attack protection at the application level, where the system blocks the Nextcloud log after a series of failed

[2] Time-based one-time password [52]

login attempts (for up to 25 seconds). In addition, Nextcloud supports Fail2ban [8], which offers protection from brute force attacks at the IP level by banning IPs that have too many failed login attempts. Fail2ban cannot be enabled from the web interface, it can only be installed and configured from the command line by connecting to the Nextcloud Server (via ssh for instance).

- **Code signing**: is a native application that performs an integrity check on Nextcloud system files. It checks all files related to system functionalities and apps and, if a match is not found, it reports an error message on the administrator's console. Note, that no integrity checks are performed on users' personal files (there is currently no functionality to provide data integrity on users' files).

- **SSE**: Server Side encryption is an application that can be enabled directly by the administrator via a browser. It offers data-at-rest protection by means of an encryption mechanism performed by the Nextcloud Server before storing files on remote storage. There are two methods that can be used for key management: 'server key' or 'per-user key'. In the first case, there will be a single private/public key pair to manage file encryption (the server key pair) while, in the second, there will be a private/public key pair for each user. The mechanism for using these keys is identical, however, with per-user keys, there is greater security. The functioning of ESS with "per-user key" will be explained in detail below.
  Nextcloud creates a 4096 bit RSA key pair for each user, both keys will be stored on the Nextcloud Server, but the private key, will be stored encrypted. Each user's private key will be encrypted with a "private key password" obtained from a "Key Derivation Function" [REFERENCE] that uses the user's password, 100,000 iterations and a salt.
  For each file, Nextcloud generates a 256-bit *file key* that will be used to encrypt that file with AES-256. The *file keys* are then also stored (in storage) in an encrypted manner using the public key of the user, who has access to that file, as the key to encrypt the *file key*. When a user wishes to access a specific file, the following steps, shown in the Figure 2.4, are performed:

  – The user's password, used for logging in, will be used to create the "private key password" (using the Key Derivation Function) with which the user's private key will be decrypted.
  – The user's private key will be stored in the "user's PHP session" (this session will be further encrypted) for as long as the user remains logged in.
  – With the private key, it will be possible to decrypt the *file key* of the file to be accessed.
  – Once the *file key* has been obtained, it will now be possible to decrypt the file and read its contents.

- **CSE**: Client Side Encryption is an asymmetric encryption mechanism that can be enabled by the administrator (by downloading the app from the web interface). This functionality is managed directly from the Desktop Client or the Mobile app. Files will be encrypted before leaving the device, so that no one, not even Nextcloud itself,

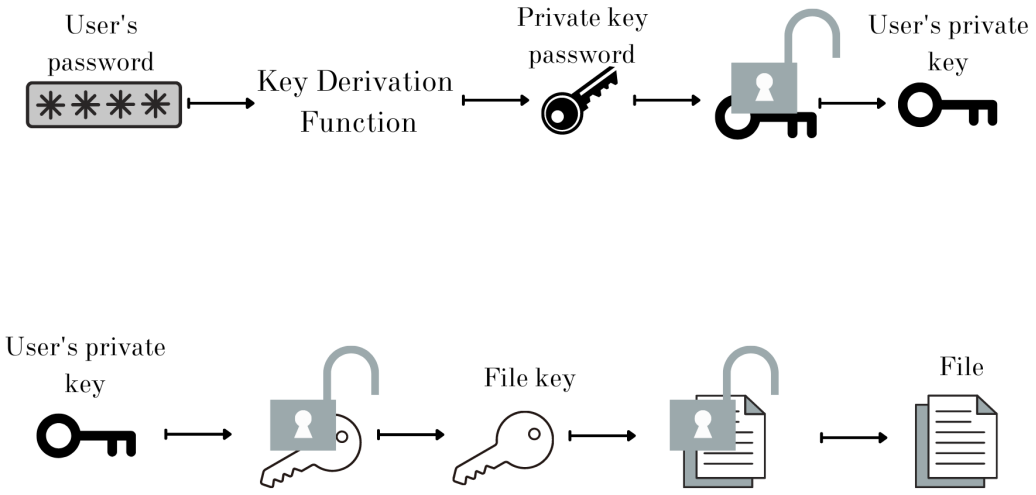

Figure 2.4. On the top, the user's password is used to obtain his private key; on the bottom, the steps to decrypt a file having the private key

will have access to the content of these files. To enable this mechanism, the files remain stored unencrypted on the user's device, and for this reason, he or she will have to take care of protecting them appropriately. When a user uses CSE, he will create an X.509 certificate with a private/public key pair. Through an asymmetric encryption mechanism, the files will be encrypted before they are even sent to the server. In addition, the private key will also be encrypted using '12 word mnemonic' (generated by the Desktop Client or Mobile app) and then, uploaded to the server. In this way, if the user connects to Nextcloud with a different device, he will simply download the encrypted private key and decrypt it with the '12 word mnemonic'. In this way, he will be able to view the contents of CSE-protected files and folders on the latter device as well.

Chapter 3

Implementation

In this chapter, different cloud storage solutions based on Nextcloud are going to be presented. In particular, prototypes V1 (3.1) and V2.0 (3.2) have been implemented by means of a virtualized infrastructure exclusively operating inside a private Local Area Network (LAN). Prototype V3.0 (3.4) has been implemented using a virtual machine (VM) and an AWS bucket as external storage. Finally, all the other proposed prototypes have not been implemented because they require dedicated hardware, implying significant costs. For this reason, the study on these prototypes is limited to an in-depth analysis from a theoretical perspective.

3.1 Prototype version 1: "Nextcloud all in one"

This solution represents the fastest and easiest method to deploy the Nextcloud platform on a completely on-premise manner. All the necessary devices (e.g., servers, PCs, etc.) are virtualized by means of Virtualbox [39]. The Nextcloud 'engine' was deployed on a Ubuntu 20.04 LTS virtual machine, since it is considered by Nextcloud developers to provide better reliability and stability than Ubuntu 22.04 LTS[30]. The virtual machine is provided with 4 GB of RAM memory and 50 GB of storage space; moreover, the VM is placed inside a private network.
This infrastructure is extremely simple because it is made of a single machine doing all the work, such as hosting the Nextcloud backend, the frontend for users, and hosting also the files stored in the cloud architecture. This simplified approach is useful to deploy Nextcloud just to test some features of the platform, and to understand its potential in terms of an open-source cloud infrastructure capable of providing security, reliability, availability, and usability.
This type of solution could be used to deploy a domestic cloud, but it cannot be used in a realistic working environment because it has flaws with respect to security and data management: it is not advisable to keep the storage together with the backend/frontend of the system, and in addition, having the files and encryption keys on the same machine involves greater risks. In order to deploy this simple Nextcloud infrastructure, one of the following methods should be used:

1. Install and configure Nextcloud on Ubuntu following the instructions of the official

documentation [26].

2. Install Nextcloud using the snap package system of Ubuntu (after installing the OS or during its installation, if the PC is connected to Internet).

3. Download from the Nextloud website an already configured '.ova' file ready to be deployd as a VM through software such as Virtualbox [25].

Method 1 should be used when it is necessary to configure parameters of the Nextcloud environment depending on specific requirements. Methods 2 and 3 offer a more 'ready-to-use' solution that might be best suited to people without the knowledge for configuring internal parameters of Nextcloud.
In Figure 3.1 the scheme of this basic Nextcloud architecture is represented. The 'Nextcloud

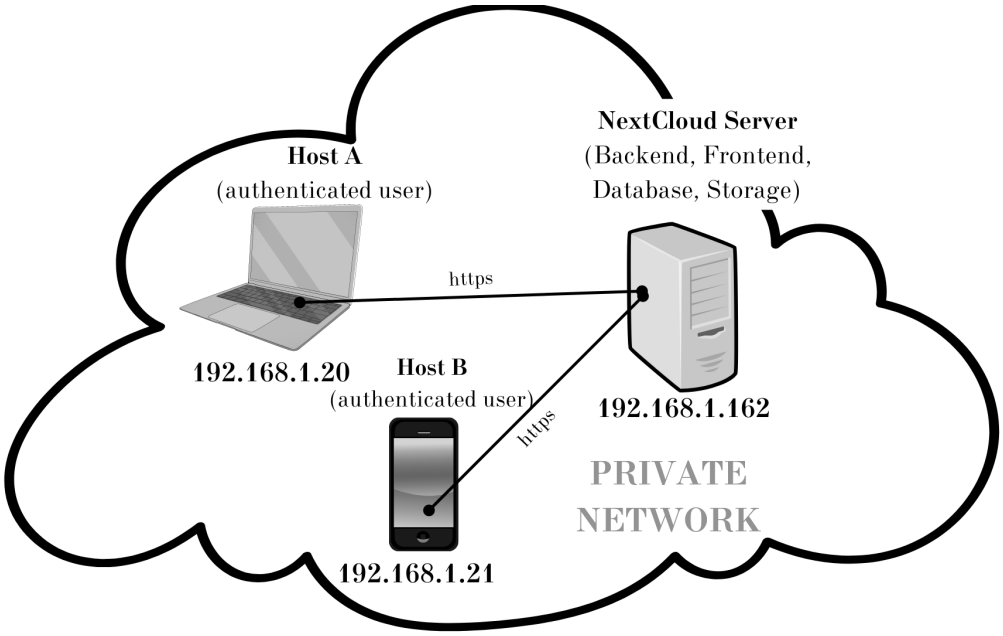

Figure 3.1. Schema of the prototype v1 *All in one*.

Server' is the virtual machine, based on Ubuntu Server 20.04 LTS, where the Nextcloud Server software (release 24.0.0) has been installed; this virtual machine is used as backend, frontend and also as data-storage. Since this prototype was tested on a private network, the IP addresses assigned to the devices involved during the test of the prototype are private (subnet 192.168.1.X/24). It is possible to access to the Nextcloud Server in two ways:

- physically, using the credentials configured during the installation of Ubuntu Server;
- remotely, by means of SSH (root login has been disabled for security reasons).

It is important to remind that credentials must be chosen carefully, it is in fact necessary to have a strong and unpredictable password that should also be changed relatively

frequently. The administrator manages the Nextcloud Server connecting to Nextcloud-ServerIPAddress/nextcloud (i.e., 192.168.1.162/nextcloud). Accessing to the Nextcloud Server, the administrator can manage the cloud platform leveraging all the features of Nextcloud.
Hosts A and B represented in Figure 3.1 are two users of Nextcloud who can access to their accounts hosted on the Nextcloud Server since they are in the same private network. In particular, a user can access to his/her account by means of a web-based interface, a desktop client, or the mobile app.
As already stated, this kind of approach deploying the entire Nextcloud platform on a single device (the Nextcloud Server) is simple but has many flaws. There are, in fact, three main tasks that must be executed: providing a frontend for users to connect to the platform, managing the platform from a low-level point of view (the backend), and managing the data storage. Aggregating all these tasks on the same machine is a problem because there is a single point of failure and it is not possible to deploy a system that is highly scalable and reliable. Therefore, this approach is suited only to very small environments (few users, small amount of data) and to testing purposes.
There are, moreover, security issues. Access to data is protected by means of HTTPS, but this only concerns data in motion (i.e., when the user access to his/her account to view or download some data). Data at rest, on the contrary, is more difficult to protect. LUKS (Linux Unified Key Setup) [18] can be used to encrypt the data on the disks belonging to the Nextcloud Server, but this is effective only when the system is powered off (i.e., disks are physically stolen), because the content of disks is decrypted (hence, it becomes accessible) during the bootstrap of the system. Additionally, Nextcloud provides a server-side-encryption feature to encrypt data but, as stated by the developers [23], it becomes useless if an attacker gains access to the machine (the Nextcloud Server) because on the very same machine there are also the cryptographic keys required to decrypt data. Because of these significant security issues, this prototype is inadequate for companies, organizations, and individuals looking for a more secure and reliable cloud storage solution.

3.2 Prototype version 2.0: "On-premise solution with a NFS share"

Prototype version 2.0 represents an evolution of the Prototype v1.0 (3.1), because it adds a new device to the scheme, using it for data storage. In fact, as represented in 3.2, there is now an additional element that is called ‘Nextcloud Storage‘. On the other hand, the Nextcloud Server does not change with respect to the previous prototype, it is a virtual machine based on Ubuntu Server 20.04.4 LTS, with Nextcloud Server software (release 24.0.0) installed.

Focusing on the Nextcloud Storage, it is a machine dedicated to store the data managed by Nextcloud (i.e., data belonging to Nextcloud users). This is convenient to separate the part of the infrastructure doing the frontend/backend work from the part responsible for storing the data. The Nextcloud Storage communicates with the Nextcloud Server via NFS share (configured to accept only the traffic coming from the Nextcloud Server). Data at-rest are protected by enabling the Nextcloud Server Side Encryption (SSE) and LUKS. In this specific experiment, the Nextcloud Storage is a virtual machine based on Ubuntu

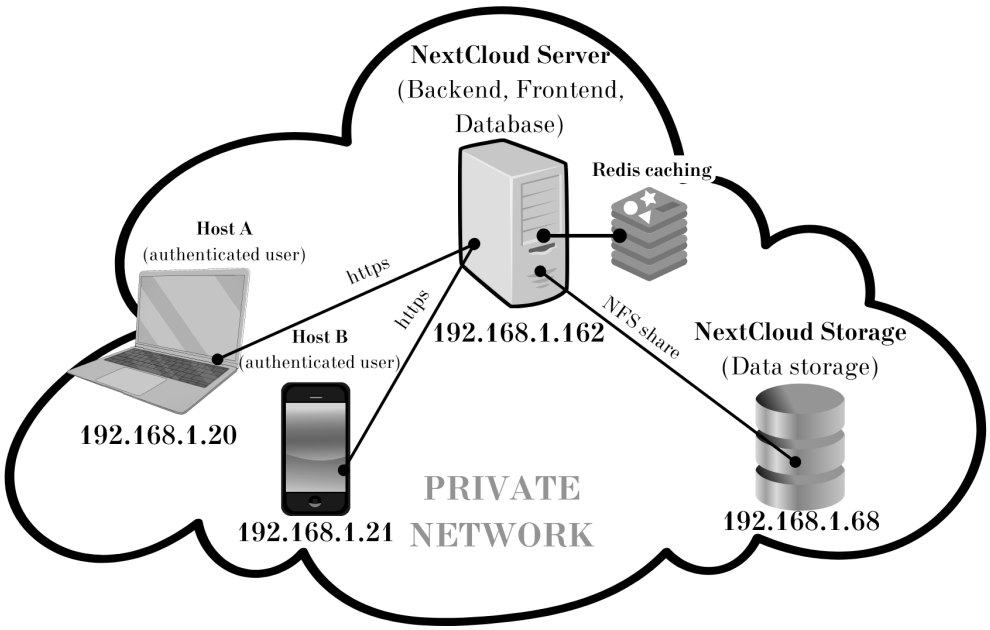

Figure 3.2. Schema of the prototype v2.0 *On-premise solution with a NFS share.*

Server 20.04 LTS. A self-signed certificate for the Nextcloud Server has been created, so that communications with clients is protected by means of HTTPS. The communication channel between the Nextcloud Server and the Nextcloud Storage, on the other hand, does not require HTTPS because the Nextcloud Server Side Encryption has been enabled. This means that all files that need to be sent to the Nextcloud Storage are encrypted by the Nextcloud Server, resulting in a system that keeps the keys on the Nextcloud Server and the encrypted data on the Nextcloud Storage (more details in Chapter 2). A REDIS caching [44] has also been installed in order to furtherly improve the performance; it works like a cache memory that stores the data in RAM memory, this provides very good write and read performance as it is not required to run the queries on the database if the requested data is already stored inside REDIS.
This second solution provides significant advantages in terms of performance, security and reliability. First of all, the Nextcloud Server is now used only to interact with users and to run the cloud engine, while a dedicated storage is responsible for managing data. This distributes the load over multiple devices and also makes the system more scalable because data storage is independent from the actual cloud engine, meaning that the storage space can be expanded without too many issues. This, in turn, makes the system more reliable because it is easier to add redundancy to the infrastructure (i.e., data backup, Nextcloud Server configuration backup, etc.).
A straightforward improvement for the Nextcloud Storage is the use of a NAS (Network Attached Storage), a storage system that usually consists in multiple hard drives installed accoring to specific RAID (Redundant Array of Independent Disks)[48] configurations in

order to boost performance, availability, and reliability.
Finally, the Nexcloud Storage is protected by LUKS encryption to protect the device's data in the case it is lost or stolen and by the Nextcloud SSE (Server Side Encryption) to prevent attackers from accessing data since they are encrypted with AES-256 (more details in Chapter 2). Unluckily the data breach problem endures if a non-authorized user gets access to the Nexcloud Server and obtains the root's permissions: when a user logs in and requests access to a file, the private key is decrypted using the user's password and stored inside the user session information. This session is always encrypted with a token provided by the client device of the user, but a sufficiently advanced, adversary with full access to the server and root's privileges would be capable of intercepting keys if it could observe the entire process and so it could read these files accessed by users (more details in Chapter 2 and on the official documentation[29]).

3.3 Prototype version 2.1: "Screened subnet schema"

This prototype is a variation of the previous one with some upgrades regarding security that make it a more suitable solution for real-world applications. Prototype v2.0 (3.2) has a significant limitation: the entire cloud platform still operates in a private network, so it is not usable by users who are not inside that network. This issue is solved with Prototype 2.1 by assigning a public IP, and a domain, the Nextcloud Server, which is now reachable through Internet. This enables users to connect to the cloud platform wherever they are, but it comes at a significant cost: exposing the cloud service to anybody, with a consequent increase to the attack surface. Because of this, it is necessary to protect the Nextcloud Server by means of firewalls, IDS/IPS, and other technologies.
The Prototype v2.1 is based on the so-called *Screened Subnet* schema [40], which is represented in Figure 3.3. The Nextcloud Server Frontend/Backend is placed in a DMZ (Demilitarized zone), while the Nextcloud Storage and Nextcloud Database are placed in a private network. We assume that all the elements of the infrastructure are physically located in a place where it is possible to implement such interconnections, otherwise it would be necessary to develop a more complex architecture to take into account, for example, that the Nextcloud Storage might be physically located somewhere else.

- Packet filter 1: it is a firewall connected to the external network (i.e., Internet) that represents the first line of defense, it analyzes incoming traffic at network level and decides if it can be permitted or must be denied.

- Packet filter 2: it is again a firewall, but it must be different from the firewall used as first packet filter. This firewall is also responsible for implementing the filter that allows inbound communication toward the Nextcloud Storage and Nextcloud Database, but only when it comes from the Nextcloud Server. The two packet filters should be sold by different vendors because if they are the same and a bug is discovered this might affect both devices and it will compromise the security.

- Gateway: this is also called bastion host and it is needed for packets that need further inspection. A bastion host stands between the DMZ and the private network and filters content exchanged between the two networks, this machine generally hosts

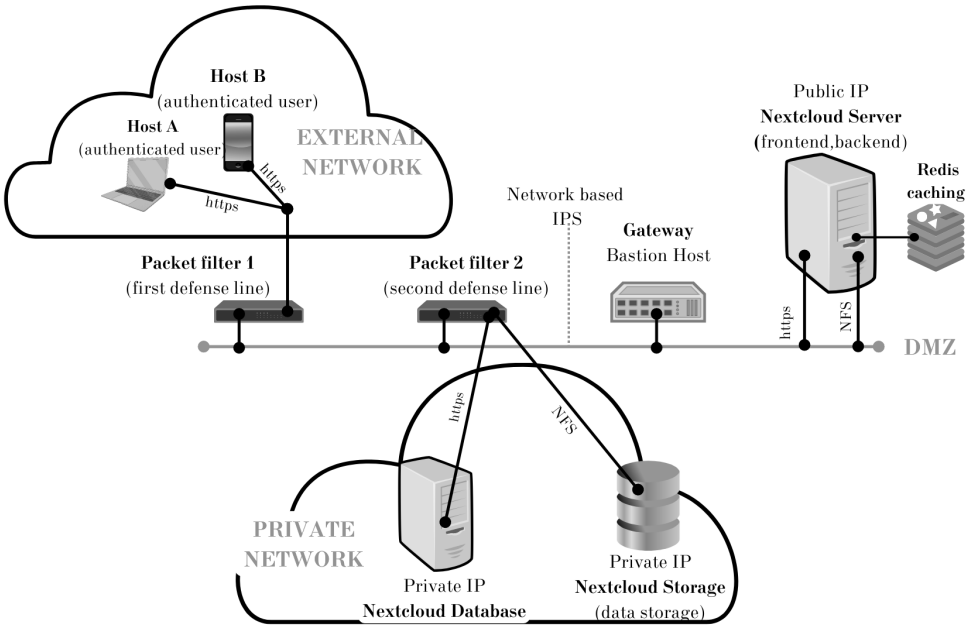

Figure 3.3. Schema of the prototype v2.1; Nextcloud Server inside the DMZ, Nextcloud Database and Storage inside the private network.

only a single application (e.g. a proxy server) because in this way it minimizes the threat of infection of the system itself through software bugs from other, non-essential services.

- NIPS: it could be really useful to have a Network-based IPS (Intrusion Prevention System) to monitor the network traffic and detect non-authorized users or authorized actors violating their privileges.

- Nextcloud Server: this server manages the frontend and backend of the system, so it is the entry point for all incoming connections. It has a public IP address, and it is placed in the DMZ to physically separate it from the data storage and datbase. This server is also equipped with a valid certificate so that connections are established only in HTTPS. Finally, this server needs two network interfaces: one for the communication over HTTPS with the clients and the Nextcloud Database, and the other one for the NFS share with the Nextcloud Storage. The Nextcloud Server having both frontend and backend functionality provides a graphical interface via Vue [53] application that shows the server response and it handles backend operations like authentication, verification, querying the database, and loading of files from the Nextcloud Storage.

- Nextcloud Database: this is the database of the system (MySQL, MariaDB or PostegreSQL), it is located inside the private LAN and it has a private IP address. The database contains user details, file's location and its sharing information and others; its task is then to provide to Nextcloud Server the information or directories of the

requested files based on the queries made. It must be reachable only by the Nextcloud Server or, equally, all inbound traffic coming from devices that are not the Nextcloud Server should be blocked. The traffic with the Nextcloud Server is encrypted with HTTPS.

- REDIS caching: Redis preserve the data in the RAM memory, saving them in a persistent way just in a second moment. If a data or information already resides on Redis, it will then be made available immediately and it will not be necessary to query the database and make the query. So a Redis cache server will reduce database load and speed up data access.

- NextCloud Storage: it only has a private IP address, and it is in a private network that is not reachable from other devices except for the Nextcloud Server. It is a remote storage that contains all the user's data and files, for this reason, data must be protected with LUKS and SSE (Server Side encryption).

This solution based on a *Screened subnet* architecture is a much more solid and sophisticated way to deploy a cloud platform in a fully on-premise fashion. If the installation and configuration of the various devices and of the security systems aforementioned is properly done, this prototype is usable by small to medium entities (i.e., a business, an organization, etc.) that want to keep the entire infrastructure under their phisical control. An on-premise installation, however, also imply issues such as dealing with hardware maintenance and managing more highly-qualified personnel.
In order to improve the system performance it is also possible to increase the Maximim Trasmission Unit. It is a parameter indicating the maximum size (in bytes) of a data packet sent via a communication protocol, the use of a larger MTU allows to send fewer larger packets to achieve the same network throughput and this means an improvement in performance. It is not advisable to modify it because of the ack that arrives from abroad and travel on HTTPS, but potentially it can be increased up to a limit of 9000 bytes against the 1500 bytes supported by standard Ethernet for the communications that occur between the Nextcloud Server and the Nexcloud Storage. Since large files and data will be transmitted in communication between the Nextcloud Server and the Nextcloud Storage, this would certainly lead to a performance improvement. Several guides are available online to determine the ideal MTU sizes based on ping test [4]. The Nextcloud server must have two network cards, one for the communications via HTTPS with MTU of 1500 bytes and the other one for the NFS share with MTU of 9000 bytes. To further improve system performance, it is recommended to use a NAS, which is a device safe in terms of data redundancy because it is protected by RAID systems, and offers speed or redundancy, but also data backup on multiple disks.

More specifically, keeping with the theme of open-source, an effective alternative could be Truenas [49], which is one of the world's most popular operating systems produced by iXsystems[14] and based on FreeBSD[10], by default it supports the OpenZFS[38] filesystem and offers advanced data management security, unlimited snapshots and checkpoint pools, data integrity, scalability, and more. At the hardware level, it is necessary to have a machine that has several hard disks, at least four, so that a RAID10[48] solution can be used, which is considered the most robust and secure because it is a combination of RAID 1 and 0 levels where several RAID 1 systems form a RAID 0 system. This offers performance

and safety against failure. TrueNAS offers several versions, the free one is called CORE[50], once downloaded it is then possible to install the TrueNAS CORE operating system on the machine. Next it is necessary to configure the NFS share so that the NAS device can be connected to the Nextcloud Server exactly as done in the v2.0 solution (3.2) obviously by entering in the mount point the IP and directory of the NFS share configured on TrueNAS device. When High Availabilty is requested, there is the possibility to upgrade from TrueNAS CORE (the costless version) to TrueNAS Enterprise with which comes a TrueNAS M-Series/X-Series [51] device which is a technology equipped with a dual controller. A device equipped with a dual controller offers high reliability because it is a device that has two different controllers inside it, which means that the moment one of them breaks down, the other one is available so that the service is always active (unless they both fail at the same time). It is therefore useful to ensure that after a failure the service continues to be up and running. So these devices offer great performance and high availabilty.

With respect to throughput, this architecture could support hundreds or few thousands of users, provided that it is backed-up by a proper network connectivity. However, there is a bottleneck represented by the fact that all network traffic must pass through various packet filters, so the upper bound is the throughput of the slowest packet filter. Furthermore, the system scalability is vertical, which mainly means that it is possible to scale up and thus improve the system by scaling only the equipment in use, e.g. by purchasing a faster processor, more powerful RAM, etc.. Whereas if it were intended to scale horizontally, this would mean having a method of growing the system based on adding new resources instead of improving the current ones. For horizontal scalability might be necessary to introduce a cluster, that is a collection of computers connected to each other via a computer network; the purpose of it is to distribute very complex processing among the various computers, increasing the system's computing power and ensuring greater service availability. This entails a higher cost and complexity of managing the infrastructure and will not be discussed in depth.

3.4 Prototype version 3.0: "Hybrid solution"

The prototypes that have been proposed in previous sections are based on fully on-premise infrastructure. While this has clear advantages, such as the physical control of the infrastructure itself, it also implies several problems related to the cost of installing, maintaining, and managing on-premise hardware. A tradeoff that might be acceptable, depending on specific requirements, is to migrate from an on-premise deployment to a remote deployment where the hardware is rented from trusted third parties (TTPs) providing, for instance, servers that are already installed in server farms with all the equipment that is required to guarantee high availability and reliability (i.e., UPS, backups, etc.).
Prototype 3.0 adopts this strategy resorting to an external provider for data storage; therefore, the 'Nextcloud Storage' is not physically located inside a private network but is hosted in some datacenter owned by a TTP. This can be done in practice purchasing a VPS (Virtual Private Server) or a dedicated, physical server from several vendors (for more details see Chapter 2).
The goal of this approach is to remove the complexity related to the choice, installation,

configuration, maintenance and management of on-premise hardware. At the same time, this advantage must not be gained at the expense of security; meaning that data must be properly protected while in-motion and at-rest. Moreover, it is fundamental to guarantee the confidentiality and integrity of data along their entire lifecycle, ensuring that the TTP (the storage provider) cannot access to the actual content of data.
As represented in 3.4, the Nextcloud Server is still deployed on-premise, meaning that the backend and the frontend of the cloud platform are hosted on a server under the physical control of the organization/company/individual owner of the platform itself.
From a technical perspective, Nextcloud supports the use of object storages [33] as primary storage location, such as Amazon Simple Storage Service (S3), OpenStack Swift, or any other compatible S3-implementation (i.e., Ceph Object Gateway, Minio). Buying some storage from TTPs such as Amazon S3 is therefore the most straightforward solution to implement a hybrid cloud infrastructure where complex tasks such has storage deployment and management are delegated to contractors capable of granting high availability, scalability and reliability.

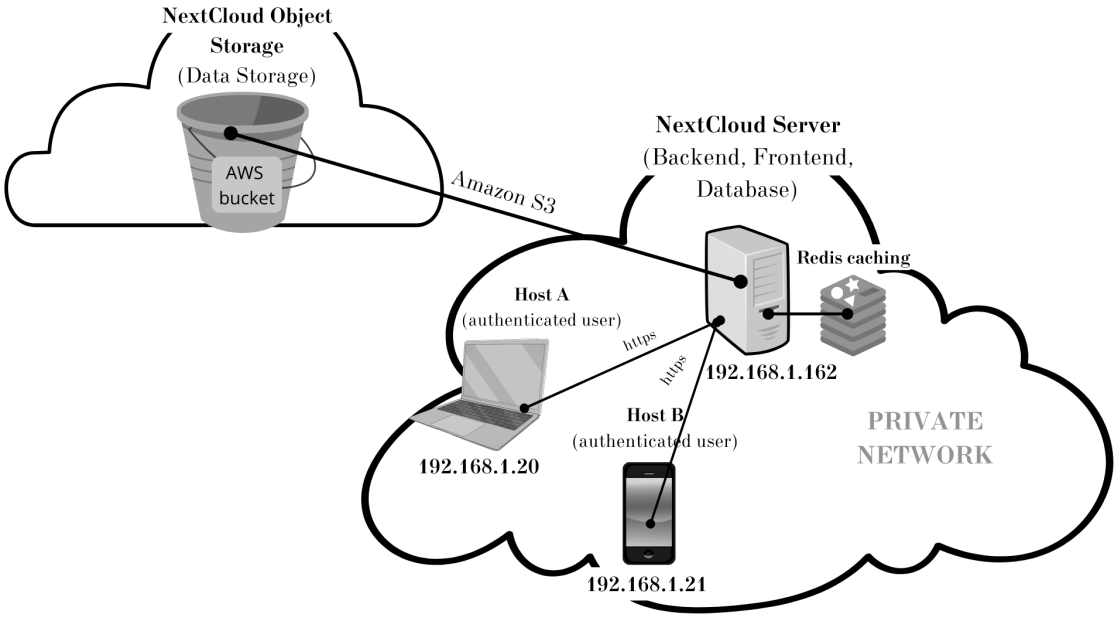

Figure 3.4. Schema of the prototype v3.0; object storage as primary storage with Amazon S3 bucket.

Prototype v3.0 has been implemented moving the data managed by Nextcloud from the server inside the private LAN to an AWS bucket. In this case, the AWS bucket plays the same role that in Prototype v2.0 was played by NFS. Similarly to the previous prototypes, the Nextcloud Server has been deployed on an Ubuntu-based virtual machine (Nextcloud 24.0 installed on Ubuntu Server 20.04 LTS).
Concerning security, data in motion between users and the Nextcloud Server is protected by means of SSL/TLS self-signed certificate was used and all the communication, inside the

private LAN, were forced on HTTPS. Moreover, the server-side user-key type encryption of Nextcloud was licensed in such a way that the content of files is encrypted by the Nextcloud Server with dedicated keys for each user before sending the files to the AWS storage (more details in Chapter 2). As already stated, the Nextcloud Storage has been moved to Amazon AWS. In particular, a free Amazon S3 plan (providing 5 GB of storage) was subscribed in order to test the prototype. Then a bucket, which is basically a container of objects, is created. Amazon S3 stores data as objects within these resources called 'buckets' and an object consists of a file and/or any metadata describing that file. Each bucket has a name and a region[1] which are decided by the client, then AWS provides a key, which is a string that uniquely identifies the bucket and a secret, another string random necessary to connect to the bucket and perform authentication. Being in possession of these data, it is possible to configure the Amazon S3 object storage as primary storage of Nextcloud by modifying and entering these parameters in the configuration file of Nextcloud [33] (specific configuration details can be found in the appendix 6).
Please note that the configuration of an external storage as primary storage location for Nextcloud should be done during the installation of Nextcloud itself; the reason being that modifying the primary storage moving from some location (i.e., an on-premise NAS) to some other location (i.e. Amazon S3) will make all the data on the first location completely inaccessible from the web interface (even though they are not physically deleted).
The problem is that Nextcloud does not support the migration of data from the default primary storage (i.e., the storage location configured during the setup of Nextcloud) to object storage (i.e., Amazon S3); therefore, data that were on the default storage even if they were not deleted will no longer be reachable by users, and the only way to allow users to access them again would be to restore the default storage as primary storage (which would make no sense).
A subscription to a storage service such as SWIFT[46] or S3[2] implies spending an amount of money that might be significant for small entities and individuals; moreover, it also implies the fact that the service provider gets physical access to data. This is not an issue if strong encryption algorithms are used, because data are sent to the object storage already encrypted and the service provider has no way to access to their value since the keys are physically stored somewhere else. There is, moreover, a concern related to business continuity; it is in fact necessary to pick a storage service provider capable of granting a top-level SLA (service-level-agreement), which includes high redundancy, reliable backups, impeccable hardware management and high scalability.
It is also possible to specify the region where the AWS bucket or the SWIFT container will be located, but it will not be possible to know specifically on which server or device they are. For these reasons, this solution is suitable for those who only want to store private files, such as personal photos and videos that do not require excessive storage space, in a place where no failures are allowed, or for a large company that can afford to incur a large cost to support all the traffic and stored data but is not interested in having accurate control over its data and files.

[1] it represents a set of data centers located in a specific geographical region of the world

3.5 Prototype version 3.1: "Nextcloud Storage on a VPS/dedicated Server"

Prototype v3.1 is an evolution of the previous one (3.4) in which traditional storage space is no longer simply bought from a TTP, but a VPS (Virtual Private Server) or a dedicated server is rented from a TTP and it is configured as the new Nextcloud Storage system. Renting a VPS or a dedicated server is a much more expensive choice than just buying some storage space, but this approach offers advantages in terms of performance and security.
A Virtual Private Server is for all intents and purposes a virtual machine running on top of a server; by purchasing such a service then you have a guaranteed level of server resources (e.g., RAM, storage space, number of cores) specified in the hosting plan, however, storage space will be shared on the physical server with other clients of the hosting company. On the other hand, a dedicated server means renting an entire physical server, so one is completely isolated and this solution provides the highest level of performance, security and flexibility, but is obviously the most expensive.
A positive aspect of this prototype is that the installation and configuration of the Nextcloud Storage is almost identical to what has been described for prototype v2.0. For example, it is possible to connect to the VPS by means of SSH and work on it exactly as it was deployed on-premise. Finally, it is also important to adopt a suitable file sharing protocol (i.e., NFS, SMB/CIFS[37], sshfs[42], etc.) to use this server as remote data storage.
For prototype v3.1, as well as for prototype v3.0(3.4), there is the important issue of having to trust a third party to supply an essential part of the infrastructure. It is important to choose the most appropriate vendor according to specific policies and regulations, such as GDPR (General Data Protection Regulation[7]).
With respect to the Italian regulations concerning cybersecurity and data sovereignty (D.L. 105 del 21 Settembre 2019 [5]), it is interesting to notice that specific public or private organizations, as well as specific companies, might be subject to what is known as National Cybersecurity Perimeter (PSNC, Perimetro di Sicurezza Nazionale Cibernetica[36]). The PSNC is a detailed body of law that defines a cybersecurity perimeter aimed at preserving the security and integrity of the Nation; therefore, specific entities must comply with these laws (i.e., entities controlling strategic sectors such as central organs, ministries, telecommunications, banking and financial systems, etc.) that impose strict rules about which kind of service providers can be used and which kind of technologies should be used. The PSNC might require to physically keep all data inside the National borders (i.e., the hardware must be installed in a facility located on Italian soil), and to use only hardware that has been certified according to the rules established by the National Center for Evaluation and Certification (CVCN, Centro di Valutazione e Certificazione Nazionale[1]).
With respect to data-in-motion, the strategies to preserve confidentiality and integrity are related to the features provided by the file sharing protocol that is adopted to communicate with the remote storage. For instance, if NFS is used, the communication is not encrypted by default. A VPN is a solution that is commonly adopted in this case, especially considering that there is a possible vulnerability concerning the SSE feature of Nextcloud (more details in Chapter 2). SSE, in fact, does not encrypt filenames and directory names, meaning that information leakage is very likely to occur is proper countermeasures are not

implemented.

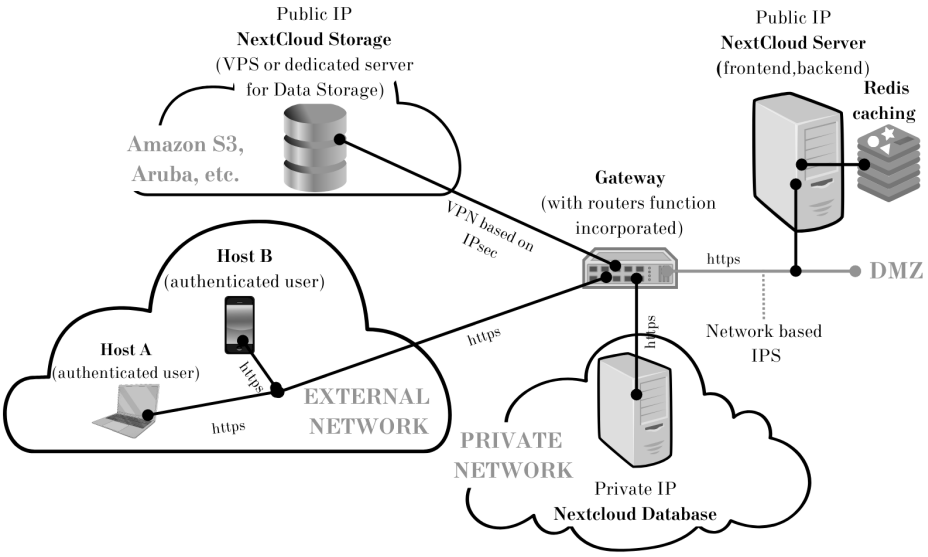

Figure 3.5. Schema of the prototype v3.1 *Nextcloud Storage on a VPS/dedicated Server*; VPN transport-mode between Nextcloud Server and Nextcloud Storage.

IPsec is a suitable choice to implement a VPN between the Nextcloud Server and the Nextcloud Storage (more information in Chapter 2). When configured in 'transport mode'(2.2.1), the IPsec VPN implements a secure end-to-end communication channel that goes from the Server to the Storage. The advantage of this configuration is that the payload is encrypted (including information needed for QoS[2], packet filtering, or intrusion detection), but the IPv4 header remains in clear since it is needed for routing.
If it was necessary to protect also the IPv4 header, a solution is represented by "End-to-end security with Basic VPN" (2.2.3); meaning that the communication channel created in 'transport mode' is encapsulated in another secure channel, the 'tunnel mode' (2.2.2) between the Nextcloud Server and the Nextcloud Storage. This approach is more secure because it also encapsulates the original packet header, but it implies more overhead in terms of time, data transmitted over the network, and use of computing resources [17].
In conclusion, prototype v3.1 represents a solution that can be used by entities having a decent IT budget, especially considering: the need of renting a VPS and/or a dedicated server, the need of buying and configuring the hardware required to setup a VPN, the expertise necessary to design and configure the overall infrastructure. In terms of scalability, this approach might very well be suited to hundreds/thousands of users.

[2]Quality of service

3.6 Prototype version 4: "Fully cloud-based infrastructure"

With this last prototype, we move completely away from any on-premise deployment migrating the entire cloud platform on VPSs or dedicated servers (2.1.2) hosted by trusted providers. This solution is clearly the most expensive, but it also offers a high degree of scalability. Because of the costs involved in the deployment of this kind of approach, it was not possible to perform any real testing of this prototype.
The implementation of this prototype is not dissimilar from what has been analyzed in the previous sections, the only real difference is that now it is necessary to access remotely to all involved servers: Nextcloud Server, Nextcloud Database, Nextcloud Storage. In order to provide high reliability and resilience at all times, it is suggested to split the structure over several servers from different providers. This means that the Nextcloud servers for frontend/backend, database and storage will need to have a clustered structure, so there will be several Nextcloud Servers , several Nextcloud Databases and several Nextcloud storage installed on different physical servers from different external providers, so that in the event of a malfunction or bug found on a specific vendor there will always be another server purchased from another vendor to ensure continuity of service. Obviously in this case the cost would be enormous, all these different servers must be synchronized, etc. so to simplify matters, in the following analyses, there will be a distinction only between Nextcloud Storage and Nextcloud Server (for frontend/backend and database functions) which will be two different servers purchased from two different vendors as it is possible to see in Figure 3.6. Concerning the protection of data-in-motion and data-at-rest, the same techniques mentioned for the previous prototypes could be adopted. The communication between the Nextcloud Server and the Nextcloud Storage, for instance, could be secured by means of a VPN (i.e., IPsec in 'transport mode' or 'tunnel mode', see 2.2.1). Data-at-rest will be protected by Nextcloud's SSE (Server Side encryption, see 2) with which the contents of all files that are sent on the remote storage will be encrypted with an AES-256 algorithm. It is important to establish with the service provider a detailed SLA (service-level-agreement) contract by which the minimum percentage of time for which the provider guarantees service delivery is determined or the average amount of time required for the provider to intervene in case of failures is defined, or the duties and responsibilities, penalties and sanctions faced by those who violate the contractual directions are defined.

This last prototype, despite being expensive, provides significant advantages because the infrastructure management is completely outsourced. There is no need to purchase, configure and maintain any hardware. There is no need for dedicated server rooms equipped with air conditioning and power supply redundancy, no need to worry about how to physically manage backups, and so on. The downside of this approach is that there is no more physical control of data and of the entire architecture, but this might be an acceptable tradeoff for entities with a significant budget that lack a skilled IT department capable of handling the low level details of the cloud infrastructure. Clearly, the specific features of the services offered by third parties must be agreed upon according to requirements such as high availability, high reliability, scalability up to a specific point, compliance with National and supra-National regulations.

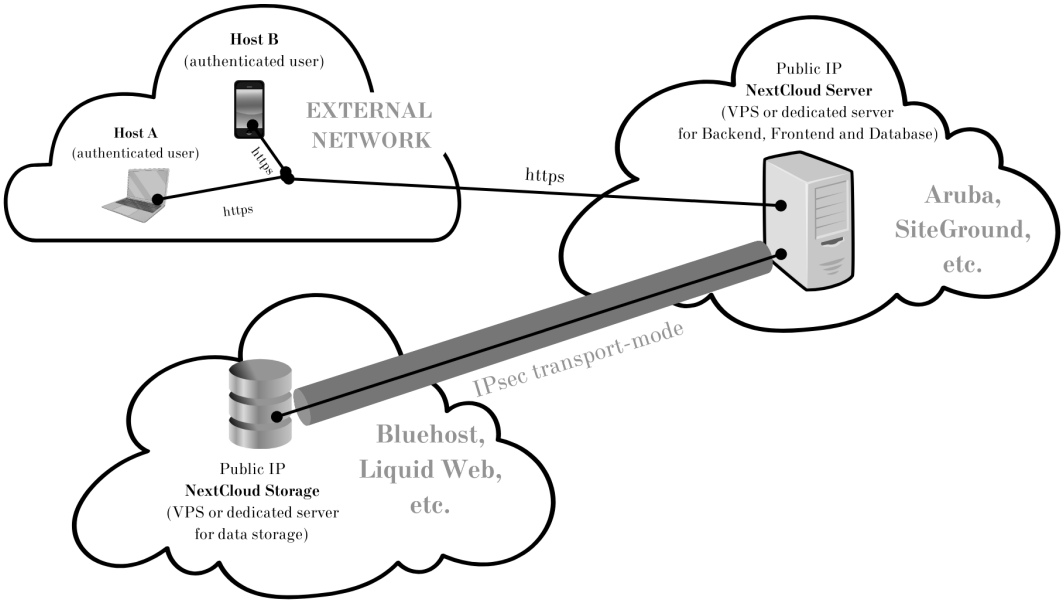

Figure 3.6. Schema of the prototype v4 *Fully cloud-based infrastructure.*

Chapter 4

Results

With respect to what has been presented in Chapter 3, the prototypes described in Sections 3.2 and 3.4 represent the two most advanced solutions that have been concretely tested during this thesis. The prototype v2.0 (see Section 3.2) can be used as a guideline to deploy a fully on-premise cloud infrastructure, while prototype v3.0 (see Section 3.4) can be used as a guideline to deploy a hybrid cloud infrastructure. Additionally, other prototypes were proposed (see Chapter 2) with the goal of improving important metrics such as scalability, reliability, and performance. However, it was not possible to test these prototypes since any deployment, even a basic one, implies a significant cost.
It must be noted that the goal of this thesis was not to address any performance-related issue, in fact the analysis was carried out in a controlled environment made of few virtual machines and a very reduced number of users (up to 4).
As mentioned in Chapter 1, the main goal of the thesis was to carry out a technology scouting in order to assess the features of Nextcloud and the feasibility of deploying it as a cloud platform capable of competing with popular commercial cloud products (i.e., Dropbox, Google Drive, etc.).

4.1 Design and schema of the on-premise and hybrid solutions

The v2.0 prototype, as shown in Figure 4.1, is composed of two virtual machines based on Ubuntu Server 20.04 LTS: one for the backend/frontend part and one for storage. The two VMs are in the same subnet with private IPv4 addresses (192.168.1.X/24). On a logical level, there is no difference if the two VMs are on the same machine, but on a technical level, it would make sense to split these two machines and put them on two different devices. This is so that if one of the two machines had a malfunction or was corrupted, this would not affect the entire system in any case. The two machines communicate via an NFS share and the SSE (Server Side Encryption) is enabled, so the contents of files are encrypted before being sent on the NFS share.
The advantages and disadvantages of this solution are:
Pros:

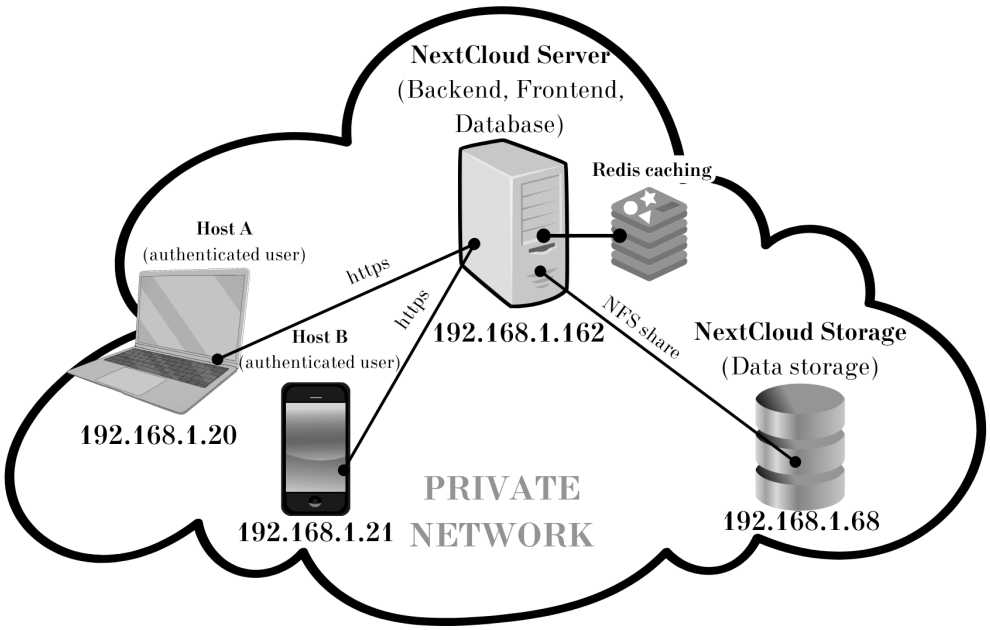

Figure 4.1. Schema of the prototype v2.0 *On-premise solution with a NFS share.*

- Appropriate security measures can be implemented to monitor and filter traffic between the Nextcloud Server and the Nextcloud Storage (e.g. configuring IDS/IPS, firewalls, etc.)
- Having the remote storage on site gives full control over data and access to the machine.

Cons:

- It is necessary to buy a specific machine for remote storage and to support all maintenance and management costs of that hardware.

The v3.0 prototype consists of a single virtual machine, the Nextcloud Server, to manage the system's frontend/backend and database, and an AWS bucket to manage the storage (the schema is shown in Figure 4.2). It is therefore a hybrid solution in which the system's storage has been moved to an external provider. In order to configure this solution, it was decided to use the Nextcloud functionality that allows an "object storage" to be used as the "primary storage", and in this case it was decided to use the service offered by Amazon S3 (for more information see the 6). The advantages and disadvantages of this solution are:

Pros:

- There is no need to buy a dedicated machine for storage or to support management and maintenance costs, it is sufficient to stipulate a subscription with the external provider.

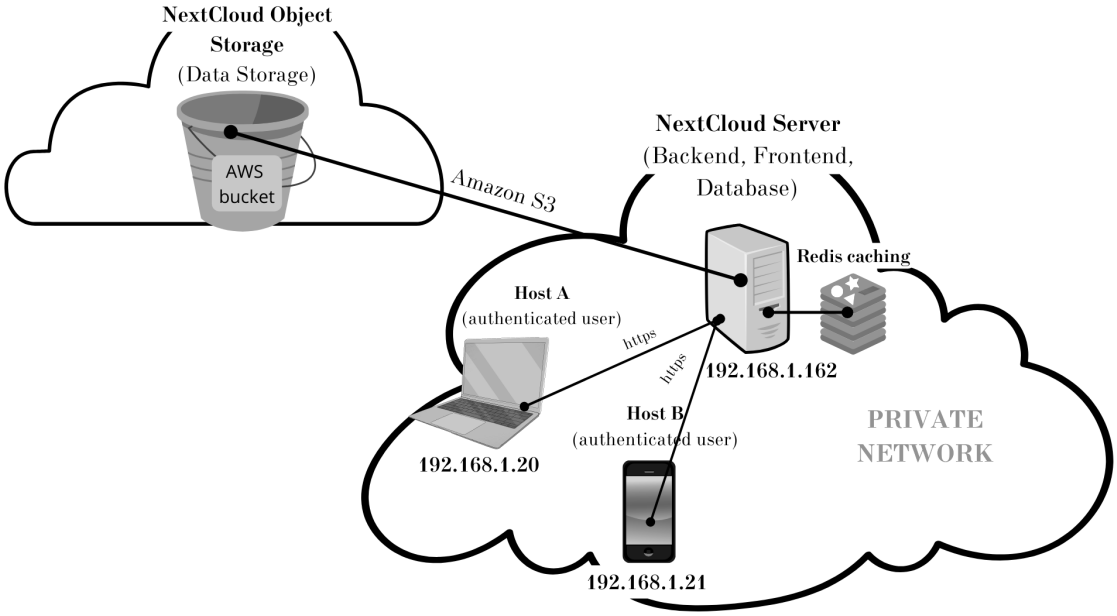

Figure 4.2. Schema of the prototype v3.0; object storage as primary storage with Amazon S3 bucket.

- It is not necessary to have a team of technical experts for data management and storage configuration, it will all be done by the external provider.
- It is easy to scale up and increase the storage space by simply paying and changing the subscription

Cons:

- Data privacy and security are completely left to the external provider, the only feasible security measure being to activate Nextlcoud's Server Side Encryption [23] (but only the content of files will be protected).

4.2 Usability and characteristics of Nextcloud

From the perspective of the users, Nextcloud provides a very simple and intuitive graphical user interface, its usage is straightforward for anybody who has already some experience with other cloud platforms providing a web-based interface such as Dropbox and Google Drive.
Similarly, also the administrator of the cloud platform can easily configure the properties and the features of the system with an intuitive web-based GUI (Graphical User Interface), even though there are few parameters that need to be configured using the CLI (Command Line Interface, see 6). Clearly, the GUI of the administrator and the GUI of the users are different, meaning that the administrator interface provides many more options to configure

the system.
Concerning the type of actors that are involved in the system, Nextcloud is very simple: there are users, and there are administrators. An administrator is also a user but, of course, a user is not necessarily an administrator. The simplest approach would therefore be to setup the cloud platform with a single administrator and multiple users, even though multiple administrators might be useful to have more redundancy and split roles in a more efficient way.
To this scheme, we also need to add the concept of group. A Nextcloud group is an entity that exists only from a logical point of view, because it is simply an attribute that is used to tag users as members of that entity (the group itself). Groups are managed by the Nextcloud administrator, but it is also possible to promote one of the members to the role of 'group administrator'. This might be useful in cases where the Nextcloud administrator wants to share some of his permissions and capabilities with one of the members. The most obvious advantage of groups is that files/folders shared with a group are automatically available to all members of that group. However, it is important to notice that the group creation and management is still a manual task; therefore, there might be a significant overhead if the specific use case requires the creation or modification of many groups in a reduced amount of time.
Access to the Nextcloud platform is possible by means of:

- web-based GUI (Google Chrome/Chromium, Mozilla Firefox, Microsoft Edge, Apple Safari)
- PC client (Windows, MacOS, Linux)
- mobile app (Android, IOS)

The recommended way of using Nextcloud is by means of the web-based GUI, because it is the only interface providing access to all features of the cloud platform. On the contrary, the PC client and the mobile app are more limited (more details in Chapter 2).
The PC client is similar to what is provided by commercial cloud services such as Dropbox. When configuring the software, a local folder named 'Nextcloud' is created; then, its content is synchronized with the Nextcloud account of the user. It is also possible to disable the synchronization of the entire content of the account, enabling the synchronization of selected files/folders only in order to save storage space and bandwidth. Files can be normally opened and modified inside the 'Nextcloud' local folder, leveraging the fact that the same files on the cloud will be rapidly synchronized with the most recent version. The PC client also shows notifications (e.g., a new file has been shared with the user); however, when the user clicks on a notification the web-based GUI is automatically opened since it is the only interface allowing the execution of the complete set of actions available for files, folders, and account settings.
Finally, the Nextcloud app for mobile devices resembles the PC client. There is the possibility to open, modify, and upload files, to locally synchronize files and folders and to check notifications. As in the case of the PC client, there are significant limitations such as the inability to configure the in-app notification system.

4.2.1 Main features

The administrator of Nextcloud can customize the cloud platform not just by means of the standard settings and features, but also thanks to ‘apps‘ that can be added to the platform. Nextcloud apps work like plugins, meaning that they add new features to the cloud storage (such as the "Talk" app for audio and video chat, "Mail" app or the "End-to-End Encryption" app for the Client Side Encryption). Considering all the possible functionalities that can be used in Nextcloud, here are the most important ones that have been configured:

- **Mail**: Nextcloud can be configured to interact with a dedicated mail server. Mails are used by the cloud platform to implement important features such as password reset and notifications; moreover, there is also the possibility to read and write e-mails using the Nextcloud web-based interface. In the specfic case of the prototypes presented in Chapter 3, a Google Mail Server [34] has been configured. Please notice that if the user wants to connect Nextcloud to Gmail, it is necessary to enable 2FA (two factor authentication) on his Gmail account and create a "password for apps" [12].
- **Notifications**: Nextcloud supports a broad range of notifications that help users in staying up-to-date with events such as sharing of a new file/folder, file modifications, etc. However, the Nextcloud administrator cannot set the notifications policy on a system-wide level, meaning that users must individually configure the notifications policy as they see fit (clearly, the cloud administrator can provide users with a guide so that they all configure the notification system according to the same rules). Notifications are delivered via email or push notification on the Nextcloud Client (PC client, smartphone app). Users can choose what type of notifications they want to receive and where (mail and/or push); notifications can be sent when editing, deleting, or adding a file/folder, for comments in messages, for changing passwords, etc. (more details in Chapter 2). The notification system has been tested on the browser, the Desktop Client and the mobile app; Desktop and mobile notification examples are shown in the Figure 4.3.
- **File Sharing**: Nextcloud provides native features to share files and folders with users and groups (if groups are enabled, which is not the case of the prototypes described in sections 3.2 and 3.4); moreover, the administrator can configure Nextcloud in order to enable file sharing with people who do not have a Nextcloud account. There is also the possibility to manage file access control (i.e. read/write permissions), file retention, and file tagging.
- **Online Office and PDF viewer**: this feature allows users to view, edit and collaborate on text files, spreadsheets, PDF files, and presentations directly within the Nextcloud web interface. However, these apps are not available on Desktop Client and mobile app. It has some slowdowns and occasional lags when several users work on the same file, but it works. It is a great way to prevent users from downloading sensitive files, in plaintext format, on their PCs.
- **REDIS caching**: Redis preserve the data in the RAM memory, saving them in a persistent way just in a second moment. This allows great performances on writing

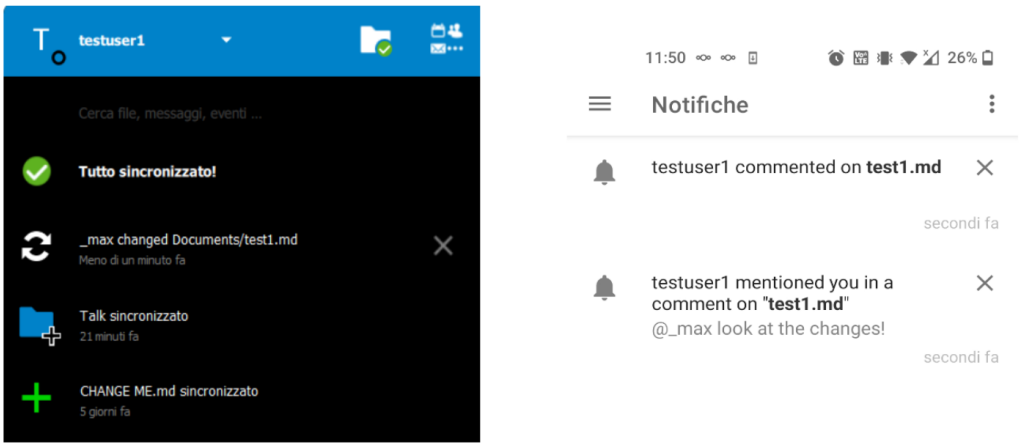

Figure 4.3. On the left, an example of notification received from the Desktop Client (Windows), on the right, from the Mobile app

and reading. It has been installed e configurated for experimental uses, in facts, to be honest, it has not been possible testing the real improvement of system's performance because of the reduced number of users in the demo realized (more information in 3).

4.2.2 Security properties

In order to provide data confidentiality, data integrity and access control across all the prototypes described (in 3.2, 3.4), Nextcloud was configured/customized adopting the technologies and protocols described below:

- **SSL/TLS**: OpenSSL has been installed, and a self-signed certificate has been created so that access to the Nextcloud infrastructure is done through HTTPS protocol (the default protocol of Nextcloud is HTTP). Notice that the use a self-signed certificate is reasonable only for testing purpose, for 'production' a certificate issued by a CA (Certification Authority) must be used.
- **2FA**: Two Factor Authentication with TOTP (Time-based One Time Password [52]) has been enabled simply by installing the application from the admin web-based interface. Users log in specifying username, password and the code generated from a TOTP app that needs to be configured in advance (see more details in 2).
- **Password Policies**: the administrator can set the password policies from the web-based interface. It has been set the minimum password length to 10, forced upper and lower case, special character and the validity window of each password it has been set to 30 days. A control to avoid the use of common passwords has been activated: the hash of the password that the user wants to set is computed, then the first five characters of the hash are sent to a website [13] that returns a list of all hashes that

begin with those first five characters, and then Nextcloud checks if the hash of the password is included in retrieved list. If there is a match, the user is asked to insert another password.

- **Fail2Ban**: it is a security tool written in Python that is designed to prevent brute force attacks [8]. It scans log files and blocks IP addresses with too many password failures, it can only be enabled if it is installed and configured from the command line by connecting, via SSH, to the Nextcloud Server.
- **Code Signing**: it is a native feature of Nextcloud which ensures that only authorized actors can push updates and that all upgrades are properly executed so that no files are left behind and all old files are replaced [21]. It performs an integrity check on Nextcloud files (excluding users' personal files), if there is some mismatch it displays a warning on the administrator console (more details in 2).
- **SSE (Nextcloud Server-Side Encryption)**: it is designed to protect the content of files stored on remote storage encrypting data-at-rest with AES-256 (more details in 2). SSE can be enabled by the administrator directly on the web-interface or with 'occ' command (installation steps are shown in detail in 6) having access to the machine via SSH. SSE can be configured to work in two ways: with a single server key or with per-user keys that are generated from the passwords of the users. For the two prototypes, 3.2 and 3.4, it has been decided to use per-user keys because they are way more secure than the single server key which, if compromised, would give an attacker instant access to all files stored by Nextcloud (for a more detailed explanation see 2). SSE has a negative impact on performance (especially when per-user keys are adopted), it also requires more storage space because encrypted files are usually 35% larger than non-encrypted files; however, the tests that were carried out (in both 3.2 and 3.4) were not able of highlighting any slowdown of the cloud infrastructure. In order to verify the effectiveness of SSE, traffic analysis and access to files directly from the storage (i.e., Amazon S3 bucket for the prototype v3.0 3.4) were carried out. Actually, the files were encrypted and it was not possible to read their contents, but since the names of files and folders were readable, it must be taken into account that there is still a minimum of information leakage. The worrying thing is that having root permissions on the remote storage, even if it is not possible to read the content of the file, it is possible to modify it. An attacker could delete all files or modify them randomly and thus compromise the entire system.
- **CSE (Client-Side Encryption / End-to-End Encryption)**: CSE is based upon asymmetric cryptography, and it can be used only through the Nextcloud Desktop Client for PC or the Nextcloud App for smartphones. As for the Server Side Encryption, also in this case it can be enabled by the administrator directly on the web-interface or with occ command having access to the machine via SSH. The idea is that data managed with the Nextcloud Client, or the smartphone app, are encrypted before leaving the device of the user, so that Nextcloud itself cannot access their content. The problem with this approach is that CSE still allows to keep plaintext data on the device of the user, which is something that is not wanted because there is no control over such device. There are also other problems with CSE, for example it is not possible to enable CSE on folders that already contain files; moreover,

files and folders that have been encrypted with this mechanism cannot be shared with other users.

4.2.3 Nextcloud scan tool

Nevertheless, as regards the system's security, all the features assumed as useful and suitable have been configurated and installed. In order to have a more solid infrastructure the native scan tool [35] of Nextcloud has been used: it is a scan that provides an overview on problems and possible improvements as regards both performance and security of the system. To consult this scan, it is necessary to log in as an administrator user to the Nextcloud's web interface, navigate to the "Settings" section and select "Overview". It is important to keep in mind that this scan does not perform any penetration test, but only a basic check, so it does not evaluate in depth the real security of the system. Despite that, in all the prototypes implemented, all errors and warnings reported by this tool have been resolved.

Chapter 5

Conclusions

The need for data storage infrastructures allowing seamless and easy access to data is constantly increasing, not just for companies and organizations but also for individuals. In response to this demand, companies such as Amazon or Dropbox offer cloud storage solutions where it is no longer necessary to worry about how to store and protect data because the provider handles all low-level details. In fact, the customer can purchase a large storage space at a reasonable cost, especially considering the high level of reliability and availability that is usually granted. Additionally, commercial cloud storage platforms provide user-friendly interfaces compatible with a plethora of devices, helping companies and organizations to reach high productivity goals. These advantages, however, come at the expense of control; when buying a cloud storage service from commercial providers, the customer hands to the service provider the physical control of data, which is not acceptable if data confidentiality is a priority. There are ways to make sure that cloud providers cannot access the content of data (e.g., only uploading encrypted files), but this significantly increases the complexity of the system and the overhead with respect to the cost to be sustained to purchase the cloud storage space. This increase in complexity is especially detrimental for the usability of the system undermining, in conclusion, the productivity and the cost-effectiveness of such cloud solutions.
This study, therefore, had the goal of investigating feasible alternatives to traditional commercial cloud storage services. The focus was set on open-source cloud storage platforms that could natively offer a user-experience and an infrastructure quality not too far from what is provided by commercial vendors. In particular, any possible alternative to be investigated was required to comply with specific features:

- File hosting services
- Open-source
- Possibility to create on-premise as well as hybrid infrastructures
- Notification system
- Possibility to collaborate on online documents
- Customizable security measures

- Data-at-rest protection
- Two factor Authenticator

Considering this list of requirements, Nextcloud (see 2 for more details) was chosen as the cloud platform to be used, since it fulfills most of the specifications.
Starting from Nextcloud, two different cloud infrastructures were implemented in order to test the feasibility of deploying an open-source cloud platform capable of competing with commercial cloud solutions. These infrastructures can be classified according to two different approaches: fully on-premise and hybrid.
Prototype 2.0, "Nextcloud with an NFS share" explored in 3.2, represents a solution consisting of two virtual machines, the first managing the backend/frontend of the system, and the second managing the storage space on a remote server. It was useful to realize and study this prototype, as it offered the possibility of understanding how to configure the various machines and how to realize and establish communication between them, in particular an NFS share. Prototype 3.0, "Hybrid solution" presented in 3.4, uses a virtual machine as in the 3.2 case for the management of the Nextcloud backend/frontend, but it uses the storage space provided by an AWS bucket through an "Amazon S3 Standard" subscription contract [2]. The experimentation carried out is therefore represented by the realization of these two prototypes. Both of them depict a very simplified solution; in fact, they have been realized on virtual machines with minimum hardware requirements and it has never exceeded four users at the same time: these conditions did not allow testing the system's real limits and performance. Furthermore, not having assigned a public IP address to the Nextcloud Server, it was not possible to evaluate the operation of the system for external users connecting remotely (tests were carried out exclusively in a private network).

Despite the intrinsic simplicity of the prototypes, their implementation has permitted a full and complete analysis of the characteristics of Nextcloud. It has also allowed to check its potential and usability. It has been very useful to see and check all the installation and configuration steps, furthermore, the customized settings according to the needs and then see how to enable all the security features mentioned before. (It has been created a complete guide concerning the installation and the implementation of the prototype v2.0 and v3.0: see the appendix 6.)

This study can be used as a starting point to develop more advanced Nextcloud-based data storage infrastructures. For instance, starting from a simple on-premise solution such as that of prototype v2.0 ("*Nextcloud with a NFS share*" 3.2), it is possible to develop and implement a more complex and evolved infrastructure such as that of prototype v2.1 "*Screened Subnet schema*" (more details in 3.3), which has the following characteristics:

- Nextcloud Server (backend/frontend) in DMZ (Demilitarized zone)
- Nextcloud Database and Nextcloud Storage in a private LAN not reachable from outside
- Firewalls to filter traffic
- IPS (Intrusion Prevention System) to detect and prevent possible threats

If a hybrid solution is desired, starting from the simplest v3.0 ("*Hybrid solution*") another example of improvement can be reached, moving to a hypothetical 3.1 "*Nextcloud Storage on a VPS/dedicated Server*" (see 3.5) solution, composed of these features:

- Nextcloud Storage on a rented VPS/dedicated server
- Nextcloud Server (backend/frontend) in DMZ
- Nextcloud Database in a private LAN not reachable from outside
- Firewalls to filter incoming/outgoing traffic to the Nextcloud Server
- IPS (Intrusion Prevention System) to detect and prevent possible threats
- IPsec-based VPN to protect NFS share traffic from Nextcloud Server to Nextcloud Storage

Analyses carried out have shown that Nextcloud is therefore a great product through which different types of infrastructures can be set up. In fact, it has a graphical interface comparable to that of other major cloud services providers, it is simple, clean and user-friendly; the system is easily customizable and it is possible to carry out most of the modifications and configurations of the system simply by connecting with an account as administrator to the Nextlcoud web interface. However, it is important to take into account that it is not a perfect solution and without problems: being open-source, bug fixes may take longer, some apps and functionalities may not meet the desired requirements.

5.1 Future work

The advisable improvements found during testing and analysis of the system are mainly improvements inherent in apps security features and storage space. Specifically, with regard to the problems concerning security features, the main issue is definitely the encryption mechanism used by the SSE (Server Side Encryption) [23]. This is an excellent feature to protect data-at-rest using a robust encryption algorithm (AES-256), the problem is that encryption only applies to the content of files; although this may not seem fundamental, it is very important. Not encrypting filenames and folder names means having an inherent information leakage in the infrastructure, because meaningful names must be used for decent usability of the system. This might not be that much of a problem for on-premise data storage, but it is a significant concern if the storage space is purchased from an external vendor. The service provider, in fact, could take advantage of this information leakage in order to have some insight about the encrypted files the customer is uploading. For this reason, it is worth considering an improvement and modification of the functioning of the SSE (Server Side Encryption) to no longer only encrypt the contents of files, but also the names of files and directories.
In addition, tests have shown in 4.2.2 that if changes are made to users' files outside of Nextcloud (i.e. by connecting with root privileges to the remote storage) the system does not detect these unauthorized changes. What happens is that such a file will no longer be readable by the user who had previously created it, and Nextcloud will report a simple malfunction and error inherent to the encryption keys. This means that not only it is

possible to make a file unusable, but also that the system is unable to detect it. For this reason, it would be advisable to modify the system functionality "Code Signing" (more details in 2), which carries out a periodic scan of the system to verify that no unauthorized changes have been made to system files and to extend it also to check the contents of users' files. Obviously, it should be configured in such a way that this check is not carried out too often, as this would entail an excessive cost in terms of performance. Another way could simply be to configure it as a specific check to be carried out only at the user's request.
Furthermore, from the point of view of system management and usability, another recommended improvement is to modify the so-called "External storage" app (more information in 2). By using this application, external storage services and devices can be mounted as secondary storage devices in Nextcloud. By enabling this application from the administrator account, it is possible in a couple of quick and easy steps to add and configure additional storage (Amazon S3, SFTP, SMB/CIFS [22]). The main problem is that this additional storage space will actually be shared among all users and anyone can see and edit the files uploaded on it. Therefore, it is advisable to modify this feature so that the "External storage" app can provide additional storage space keeping it divided among the various users, so that each user has his/her own reserved space on this secondary storage.

Nextcloud certainly offers a possible alternative to the famous big cloud service vendors, but it must be taken into consideration that if the objective is to move away from paid solutions and obtain the highest level of data control and system security, it is necessary to implement at the very least the improvements to the system mentioned in 5.1. It is advisable to take the correct protection measures, and to thoroughly analyze both the situation and the infrastructure model, so as to be able to configure, as appropriately as possible, all those additional system functions such as VPN, firewalls, IDS/IPS. In the case of on-premise solutions, the required hardware should be identified. Whereas, for hybrid or completely cloud-based solutions, it is necessary to carefully evaluate the external provider from which a VPS or dedicated server, that is able to meet all the requirements, can be purchased (e.g. GDPR, a precise and well-defined Service Level Agreement). In conclusion, at the end of this thesis, it is possible to state that Nextcloud, with the necessary improvements and refinements, can represent, in all respect and purposes, a solution that can be used by small and medium-sized companies for the implementation of a solid and functional open source infrastructure, capable of providing a cloud storage service, comparable with the most used commercial cloud services.

Chapter 6

Appendix

To avoid any misunderstandings, it is important to note that this is not the official Nextcloud installation guide; it is a guide created by the author of this thesis (Massimo Missio). In this chapter, the installation and configuration guide for Nextcloud 24.0.0 will be illustrated to build prototype v2.0 *On-premise solution with a NFS share* 3.2 and prototype v3.0 *Hybrid solution* 3.4.

6.1 NextCloud Server initial installation and configuration

Install Ubuntu Server 20.04 LTS (in this case 20.04.4 version has been choosed). The recommended RAM is 512 MB, but it has been used 2048 MB and dedicated 40 GB of disk space, during the installation of Ubuntu Server on a Virtual Machine (like VirtualBox in this example) remember to set, under network settings, the "network bridge" on, so the Ubuntu server can be reached from any device in the private network. Now update and install the recommended packages, first add the repository Ondrej [15] in order to install PHP 8.0 (Ubuntu 20.04 doesn't have this repository so it is necessary to manually add it).

```
sudo apt update
sudo apt install lsb-release ca-certificates
   apt-transport-https software-properties-common -y
sudo add-apt-repository ppa:ondrej/php
sudo apt install php8.0
```

Install the correct version of MariaDB (10.5):

```
wget https://downloads.mariadb.com/MariaDB/mariadb_repo_setup
curl -LsS https://r.mariadb.com/downloads/mariadb_repo_setup
| sudo bash -s -- --mariadb-server-version="mariadb-10.5"
sudo chmod +x mariadb_repo_setup
./mariadb_repo_setup --mariadb-server-version="mariadb-10.5"
```

Now that the correct versions are installed, install all the other recommended packages:

```
sudo apt install apache2 mariadb-server libapache2-mod-php8.0
sudo apt install php8.0-gd php8.0-mysql
sudo apt install php8.0-curl php8.0-mbstring
sudo apt install php8.0-intl php8.0-gmp php8.0-bcmath
sudo apt install php8.0-xml php8.0-imagick php8.0-zip
sudo service apache2 restart
```

Now create a database user and the database itself by using the MySQL command line interface.

```
sudo mysql_secure_installation
```

When it asks to enter the MariaDB root password, press Enter key as the root password is not set yet. Then enter Y to set the root password for the MariaDB server. Remember to give the MariaDB root user a strong password. Next, press Enter to answer all remaining questions, remove anonymous users, disable remote root login, and remove the test database. Replace *username* and *password* with appropriate values.

```
sudo mysql -u root -p
CREATE USER 'username'@'localhost' IDENTIFIED BY 'password';
CREATE DATABASE IF NOT EXISTS nextcloud
   CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;
GRANT ALL PRIVILEGES ON nextcloud.* TO 'username'@'localhost';
FLUSH PRIVILEGES;
quit;
```

Download and unzip NextCloud 24.0.0.

```
sudo wget https://download.nextcloud.com/server
   /releases/nextcloud-24.0.0.zip
sudo apt install unzip
sudo unzip nextcloud-24.0.0.zip -d /var/www/html/
sudo mkdir -p /var/www/html/nextcloud/data
```

6.1.1 NextCloud Storage installation and configuration

This subsection is dedicated exclusively to the realisation of the Nextcloud Storage for the v2.0 *On-premise solution with a NFS share* (3.2). If you are interested in realising the v3.0 *Hybrid Solution* (3.4), skip this entire subsection.

Before finishing the configuration of the NextCloud Server it is necessary to set up the NFS share, so another machine is needed. This machine, the "NextCloud Storage", is the data storage of the NextCloud system, so it could be a NAS, a Server with RAID configuration, etc. In this case, to simplifying the architecture, another Ubuntu Server installed on a virtual machine (Ubuntu Server 20.04) has been used. It must be located in the same private network of the NextCloud Server.
On a fresh Ubuntu Server 20.04 LTS installation (with 2048 MB of RAM and 40 GB of

disk space) run this command to install the NFS kernel server and create the NFS export directory.

```
sudo apt update
sudo apt install nfs-kernel-server
sudo mkdir -p /home/nfs_share
sudo chown -R www-data:www-data /home/nfs_share/
sudo chmod 777 /home/nfs_share/
```

Now Grant NFS Share Access to Client Systems.

```
sudo nano /etc/exports/
```

Add, at the end of the file, this single line. Where 192.168.X.Y is the IP of the NextCloud Server, *uid* and *gid* are the user and group id of "www-data"

```
/home/nfs_share 192.168.X.Y(rw,sync,no_subtree_check,
  all_squash,anonuid=uid,anongid=gid)
```

Export the NFS Share directory and allow NFS access through the Firewall.

```
sudo exportfs -a
sudo systemctl restart nfs-kernel-server
sudo ufw enable
sudo ufw status
```

Now go back to the NextCloud Server (192.168.X.Z is the IP of the NextCloud Storage).

```
sudo apt install nfs-common
sudo mount 192.168.X.Z:/home/nfs_share
  /var/www/html/nextcloud/data
sudo chown -R www-data:www-data /var/www/html/nextcloud/
```

The NFS share is now correctly mounted.

6.1.2 AWS bucket configuration

This subsection is dedicated to the configuration of "object storage" as "primary storage" to obtain the v3.0 *Hybrid Solution* (3.4). If you are interested in realizing the v2.0 *On-premise solution with a NFS share* (3.2), skip this entire subsection.
To use 'object storage'[1] as primary storage for Nextcloud and then to move users' files from Nextcloud's default internal directory to external remote storage, the guide provided in the official documents was followed [33]: it has been enough, after having obtained the access credentials, key, secret, and identified the region of the bucket (in this specific case has been chosen central Europe, "eu-central.1"), to substitute these values in the configuration's file 6 to enable the use of the system. As an "object storage" it has been decided to use the

[1]It must be remembered that it is also possible to configure this object storage at a later date, but all user files previously stored on the default storage will no longer be reachable.

Simple Storage Service (S3) [2] due to its great reliability and, also, because it was free. To be more specific, it has been purchased the free plan of AWS that allows an S3 Standard Amazon service for twelve months, with five GigaByte storage space, twenty thousand GET requests, two thousand POST/PUT/LIST/COPY requests and until hundred GigaByte of data transmission by month. Now run this command:

```
sudo nano /var/www/html/nextcloud/config/config.php
```

Add these lines in the config.php file (substitute *bucket*, *key*, *secret*, *region* and *port* with appropriate values):

```
'objectstore' => [
  'class' => '\\OC\\Files\\ObjectStore\\S3',
  'arguments' => [
      'bucket' => 'nextcloud',
      'autocreate' => true,
      'key'    => 'EJ39ITYZEUH5BGWDRUFY',
      'secret' => 'M5MrXTRjkyMaxXPe2FRXMTfTfbKEnZCu+7uRTVSj',
      'hostname' => 'example.com',
      'port' => 1234,
      'use_ssl' => true,
      'region' => 'optional',
      'use_path_style'=>true
  ],
],
```

Now the AWS bucket is correctly configured, files and user data will be stored on it.

6.2 Nextcloud Server configuration conclusion

The NextCloud installation is ready to finish, open a web browser and point it to http://nextcloud-server-ip/nextcloud/ to complete the final steps:

1. Enter your Username and Password to create an admin account.
2. Specify the Data folder. In this case it is /var/www/html/nextcloud/data.
3. Provide database connection settings (username, password, and database name).
4. Click the Finish setup button to complete the installation of Nextcloud on Ubuntu.

6.3 Adding Features

This section will show the installation and configuration steps for the most important Nextcloud features.

Enable Server side encryption with per-user key

```
sudo -u www-data php /var/www/html/nextcloud/occ
   app:enable encryption
sudo -u www-data php /var/www/html/nextcloud/occ
   encryption:status
sudo -u www-data php /var/www/html/nextcloud/occ
   encryption:enable
sudo -u www-data php /var/www/html/nextcloud/occ
   encryption:list-modules
sudo -u www-data php /var/www/html/nextcloud/occ
   encryption:disable-master-key
sudo -u www-data php /var/www/html/nextcloud/occ
   encryption:encrypt-all
```

Create a self-signed certificate and force HTTPS

First create a self-signed certificate, obviously it might be better to use a real certificate, Let's Encrypt [16] gives the possibility to create a free and real SSL/TLS certificate (but you need a domain name). In this case it has been used a self-signed one.

```
sudo apt install openssl
sudo a2enmod ssl
sudo mkdir /etc/apache2/ssl
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048
   -keyout /etc/apache2/ssl/nextcloudserver.key
   -out /etc/apache2/ssl/nextcloudserver.crt
```

Compile the certificate and then run this command to edit the 000-default.conf file:

```
sudo nano /etc/apache2/sites-available/000-default.conf
```

Now, change `<VirtualHost *:80>` in `<VirtualHsost *:443>` and then add these lines at the top of file (192.168.X.Y is the IP of the NextCloud Server):

```
<VirtualHost *:80>
      Redirect "/" https://192.168.X.Y/
</VirtualHost>
```

Then add also these two lines for the SSL certificate and Key file at the bottom of the file (before closing `</VirtualHost>`):

```
<VirtualHost *:443>
  .
  .
  .
  .
  SSLCertificateFile /etc/apache2/ssl/nextcloudserver.crt
  SSLCertificateKeyFile /etc/apache2/ssl/nextcloudserver.key
</VirtualHost>
```

Now HTTP is forced to HTTPS, restart apache to enable the changes:

```
sudo service apache2 restart
```

Mail server configuration with Gmail (for password reset and notification)

- Use or create a gmail account and habilitate Two Factors Authentication.
- Go under security section and create a "password for apps" [12].
- Log in with the admin account of Nextcloud, go under "Settings", then "basic settings" and compile the email server:
  - SMTP , Encryption: SSL/TLS
  - From address: example@gmail.com
  - Authentication method: Login , Authentication required: true
  - Server address: smtp.gmail.com , port: 465
  - Credentials: example@gmail.com , psw: the "password for apps"[12] generated.
  - If it has not yet been done, add a personal email from your personal information in the Nextcloud account (under Settings -> Personal info -> Email).
  - Now verifies the correct work of the mail server sending the test mail in the email server section (you will receive in your personal email a message from Nextcloud).

Mail app configuration for Gmail
These are the step to configure a Gmail account as personal mail account of Nextcloud.

- Admin need to log in NextCloud, go under Apps -> Social & Communication and download and enable the Mail app.
- Now all the users, admin included, can configure their private mail, it will appear in the top bar of the page a mail icon.
- Select the mail icon and choose the manual configuration (a Gmail account with 2FA enabled and "password for apps"[12] is needed).
  - Name: your username, Mail Address: yourgmailaccount@gmail.com
  - IMAP Host: imap.gmail.com
  - IMAP Security: SSL/TLS
  - IMAP Port: 993
  - IMAP User: yourgmailaccount@gmail.com
  - IMAP Password: password for apps generated
  - SMTP Host: smtp.gmail.com
  - SMTP Security: SSL/TLS
  - SMTP User: yourgmailaccount@gmail.com
  - SMTP Password: password for apps generated

Fail2Ban Installation

Fail2ban offers a protection, at IP level, from brute force attack. On the Nextcloud Server run these commands:

```
sudo apt install fail2ban -y
sudo systemctl enable fail2ban
sudo systemctl start fail2ban
```

Now edit the file nextcloud.conf:

```
sudo nano /etc/fail2ban/filter.d/nextcloud.conf
```

Copy and paste the following lines:

```
[Definition]
_groupsre = (?:(?:,?\s*"\w+":(?:"[^"]+"|\w+))*)
failregex = ^\{%(_groupsre)s,?\s*"remoteAddr":
  "<HOST>"%(_groupsre)s,?\s*"message":"Login failed:
^\{%(_groupsre)s,?\s*"remoteAddr":
  "<HOST>"%(_groupsre)s,?\s*"message":
  "Trusted domain error.
datepattern = ,?\s*"time"\s*:\s*"%%Y-%%m-%%d
  [T ]%%H:%%M:%%S(%%z)?"
```

Now edit the file nextcloud.local:

```
sudo nano /etc/fail2ban/jail.d/nextcloud.local
```

Copy and paste these lines, where *findtime* is the window of time (in seconds) that fail2ban will pay attention to when looking for repeated failed authentication attempts; *maxretry* is the maximum number of attempts and *bantime* (in seconds) represents the ban time for an IP.:

```
[nextcloud]
backend = auto
enabled = true
port = 80,443
protocol = tcp
filter = nextcloud
maxretry = 3
bantime = 86400
findtime = 43200
logpath = /var/www/html/nextcloud/data/nextcloud.log
```

Now restart fail2ban in other to update the changes:

```
sudo systemctl restart fail2ban
sudo fail2ban-client status nextcloud
```

Enabling Two factors authenticator

- Login with admin account in Nextcloud
- Go under Apps -> Security and download and enable the Two-Factor TOTP Provider app.
- Go to Settings-> Personal -> security and generate a backupcode (and save it), then enable the TOTP, you will need an external TOTP app.
- Then go to Settings -> Administrator -> security and enable: Enforce 2-factor authenticator.
- Now it will be enabled for ALL users.

Setting Cron Job
This is done to switch from Ajax to Cron, with a Cron Job a background job can be set periodically (in this case it has been set every 5 minutes).

```
sudo crontab -u www-data -e
```

If asked, press 1, then add at the end of the file these lines:

```
*/5  *  *  *  * php -f
   /var/www/html/nextcloud/cron.php
sudo -u www-data php
   /var/www/html/nextcloud/occ background:Cron
```

Enable Client Side Encryption (E2EE)
These are the steps to enable CSE (more details in 2), which is considered as "end-to-end security".

- Login with admin account in Nextcloud
- Go under Apps -> Security and download and enable the *End-to-End Encryption* app.
- From now all users at their next login in a Desktop or Mobile app can use Client side encryption to encrypt their folders and files.
- Remember that it is possible to enable E2EE only on an empty folder.

Bibliography

[1] ACN. *CVCN - Agenzia per la Cybersicurezza Nazionale*. https://www.acn.gov.it/agenzia/articolazioni/cvcn. [Online; accessed 30-September-2022]. 2022.

[2] Amazon. *Archiviazione di oggetti cloud - Amazon S3*. https://aws.amazon.com/it/s3/. [Online; accessed 10-September-2022]. 2022.

[3] bit2meACADEMY. *What is Zero-Knowledge Protocol*. https://academy.bit2me.com/en/zkp-zero-knowledge-protocol/. [Online; accessed 30-September-2022]. 2022.

[4] Comparitech. *How to Determine MTU Size Using ping*. https://www.comparitech.com/net-admin/determine-mtu-size-using-ping/. [Online; accessed 29-September-2022]. 2022.

[5] Consiglio dei Ministri. *D.L. 105 del 21 settembre 2019*. https://www.gazzettaufficiale.it/eli/id/2019/09/21/19G00111/sg. [Online; accessed 29-September-2022]. 2019.

[6] Digicert. *What is SSL, TLS and HTTPS*. https://www.websecurity.digicert.com/security-topics/what-is-ssl-tls-https. [Online; accessed 29-September-2022]. 2022.

[7] European Union. *General Data Protection Regulation*. https://gdpr-info.eu/. [Online; accessed 30-September-2022]. 2022.

[8] Fail2ban. *Fail2ban*. https://www.fail2ban.org/. [Online; accessed 10-September-2022]. 2022.

[9] ForcePoint. *What is Defense in Depth?* https://www.forcepoint.com/cyber-edu/defense-depth. [Online; accessed 30-September-2022]. 2022.

[10] FreeBSD. *The FreeBSD Project*. https://www.freebsd.org/. [Online; accessed 30-September-2022]. 2022.

[11] Bill Gates. *The Road Ahead*. Viking, 1995.

[12] Google. *Sign in with App Passwords*. https://support.google.com/accounts/answer/185833?hl=en. [Online; accessed 10-September-2022]. 2022.

[13] HIBP. *Have I Been Pwnded*. https://haveibeenpwned.com/. [Online; accessed 10-September-2022]. 2022.

[14] iXsystems, Inc. *iXsystems, Inc - Enterprise Storage and Servers Driven by Open Source*. https://www.ixsystems.com/. [Online; accessed 30-September-2022]. 2022.

[15] Launchpad. *The main PPA for supported PHP versions*. https://launchpad.net/~ondrej/+archive/ubuntu/php. [Online; accessed 30-September-2022]. 2022.

[16] Let's Encrypt. *Let's Encrypt*. https://letsencrypt.org/. [Online; accessed 10-September-2022]. 2022.

[17] Lukas Osswald, Marco Haeberle, and Michael Menth. *Performance Comparison of VPN Solutions*. https://core.ac.uk/download/pdf/322886318.pdf. [Online; accessed 29-September-2022]. 2022.

[18] Mattia Zignale. *How LUKS works with Full Disk Encryption in Linux*. https://infosecwriteups.com/how-luks-works-with-full-disk-encryption-in-linux-6452ad1a42e8. [Online; accessed 29-September-2022]. 2022.

[19] Nextcloud. *About - Nextcloud*. https://nextcloud.com/about/. [Online; accessed 9-September-2022]. 2022.

[20] Nextcloud. *All the applications - App Store - Nextcloud*. https://apps.nextcloud.com/. [Online; accessed 30-September-2022]. 2022.

[21] Nextcloud. *Code signing*. https://docs.nextcloud.com/server/latest/admin_manual/issues/code_signing.html. [Online; accessed 29-September-2022]. 2022.

[22] Nextcloud. *Configuring External Storage (GUI)*. https://docs.nextcloud.com/server/latest/admin_manual/configuration_files/external_storage_configuration_gui.html. [Online; accessed 29-September-2022]. 2022.

[23] Nextcloud. *Encryption configuration - Nextcloud latest administrator*. https://docs.nextcloud.com/server/latest/admin_manual/configuration_files/encryption_configuration.html. [Online; accessed 29-September-2022]. 2022.

[24] Nextcloud. *File Sharing - Nextcloud documentation*. https://docs.nextcloud.com/server/latest/user_manual/en/files/sharing.html. [Online; accessed 30-September-2022]. 2022.

[25] Nextcloud. *Index of /aio-vm*. https://download.nextcloud.com/aio-vm/. [Online; accessed 30-September-2022]. 2022.

[26] Nextcloud. *Install on Linux - Nextcloud*. https://docs.nextcloud.com/server/latest/admin_manual/installation/source_installation.html. [Online; accessed 29-September-2022]. 2022.

[27] Nextcloud. *Nextcloud - Online collaboration*. https://nextcloud.com/. [Online; accessed 30-September-2022]. 2022.

[28] Nextcloud. *Office - Nextcloud documentation*. https://docs.nextcloud.com/server/24/admin_manual/office/index.html. [Online; accessed 30-September-2022]. 2022.

[29] Nextcloud. *Server-side-Encryption-Whitepaper-WebVersion-072018.pdf*. https://nextcloud.com/media/wp135098u/Server-side-Encryption-Whitepaper-WebVersion-072018.pdf. [Online; accessed 29-September-2022]. 2022.

[30] Nextcloud. *System requirements - Nextcloud documentation*. https://docs.nextcloud.com/server/latest/admin_manual/installation/system_requirements.html. [Online; accessed 10-September-2022]. 2022.

[31] Nextcloud. *Two-factor authentication - Nextcloud documentation*. https://docs.nextcloud.com/server/latest/admin_manual/configuration_user/two_factor-auth.html. [Online; accessed 30-September-2022]. 2022.

[32] Nextcloud. *User password policy - Nextcloud documentation*. https://docs.nextcloud.com/server/latest/admin_manual/configuration_user/user_password_policy.html. [Online; accessed 30-September-2022]. 2022.

[33] Nextcloudd. *Configuring Object Storage as Primary Storage - Nextcloud documentation*. https://docs.nextcloud.com/server/latest/admin_manual/configuration_files/primary_storage.html. [Online; accessed 10-September-2022]. 2022.

[34] Nextcloudd. *Email - Nextcloud documentation*. https://docs.nextcloud.com/server/latest/admin_manual/configuration_server/email_configuration.html. [Online; accessed 10-September-2022]. 2022.

[35] Nextcloudd. *Nextcloud Security Scan*. https://scan.nextcloud.com/. [Online; accessed 10-September-2022]. 2022.

[36] NttData. *Perimetro Nazionali di Sicurezza Cibernetica*. https://it.nttdata.com/insights/blog/perimetro-sicurezza-nazionale-cibernetica. [Online; accessed 30-September-2022]. 2022.

[37] OpenSkill. *CIFS/SMB Protocol*. https://openskill.info/topic.php?ID=29. [Online; accessed 30-September-2022]. 2022.

[38] OpenZFS. *OpenZFS*. https://openzfs.org/wiki/Main_Page. [Online; accessed 30-September-2022]. 2022.

[39] Oracle. *Download - Oracle VM VirtualBox*. https://www.virtualbox.org/wiki/Downloads. [Online; accessed 29-September-2022]. 2022.

[40] O'Reilly. *Screened Subnet Architectures*. https://docstore.mik.ua/orelly/networking_2ndEd/fire/ch06_03.htm. [Online; accessed 30-September-2022]. 2022.

[41] Owncloud. *ownCloud - share files and folders*. https://owncloud.com/. [Online; accessed 30-September-2022]. 2022.

[42] RedHat. *SSHFA: Mounting a remote files systems over SSH*. https://www.redhat.com/sysadmin/sshfs. [Online; accessed 30-September-2022]. 2022.

[43] RedHat. *What is a Hypervisor*. https://www.redhat.com/en/topics/virtualization/what-is-a-hypervisor. [Online; accessed 29-September-2022]. 2022.

[44] Redis. *Redis*. https://redis.io/. [Online; accessed 30-September-2022]. 2022.

[45] StreamDatacenter. *Server Farm - Server Cluster or Collection of Servers*. https://www.streamdatacenters.com/glossary/server-farm/. [Online; accessed 29-September-2022]. 2022.

[46] Swift. *Homepage | SWIFT*. https://www.swift.com/. [Online; accessed 30-September-2022]. 2022.

[47] TechTarget. *What is Advanced Encryption Standart (AES)?* https://www.techtarget.com/searchsecurity/definition/Advanced-Encryption-Standard. [Online; accessed 29-September-2022]. 2022.

[48] TechTarget. *What is RAID 10 (RAID 1+0)*. https://www.techtarget.com/searchstorage/definition/RAID-10-redundant-array-of-independent-disks. [Online; accessed 29-September-2022]. 2022.

[49] TrueNAS. *TrueNAS - Welcome to the Open storage era*. https://www.truenas.com/. [Online; accessed 30-September-2022]. 2022.

[50] TrueNAS. *TrueNAS CORE*. https://www.truenas.com/truenas-core/. [Online; accessed 30-September-2022]. 2022.

[51] TrueNAS. *TrueNAS Enterprise*. https://www.truenas.com/truenas-enterprise/. [Online; accessed 29-September-2022]. 2022.

[52] Twilio. *What is a time-based one-time password (TOTP)?* https://www.twilio.com/docs/glossary/totp. [Online; accessed 29-September-2022]. 2022.

[53] VueJS. *Vue.js - the progressive javascript framework*. https://vuejs.org/. [Online; accessed 29-September-2022]. 2022.

Ringraziamenti

Ringrazio il Prof. Paolo Ernesto Prinetto, mio relatore, per la professionalità e per avermi concesso la possibilità di lavorare a questo progetto.

Un ringraziamento particolare al Dott. Matteo Fornero e al Dott. Nicolò Maunero, per i validi e fondamentali consigli e la costanza con la quale mi avete sostenuto e guidato in questo percorso.

Alla mia famiglia, la mia fidanzata, i miei amici e compagni di università, grazie per essere stati miei complici, ognuno a suo modo, in questo percorso intenso ed entusiasmante, vi voglio bene.