Haaga-Helia
University of Applied Sciences

Cloud Security: Private Cloud Solution with End-to-end Encryption

Trinh Ngo

Bachelor's Thesis
Degree Programme in Business IT
2018

Haaga-Helia

Abstract

21 May 2018

Author(s)
Trinh Ngo

Degree programme
Bachelor's Degree in Business Information Technology (BITE15S)

Report/thesis title Cloud Security: Private Cloud Solution with End-to-end Encryption	**Number of pages and appendix pages** 61 + 32

The main aim of this thesis paper is to become knowledgeable about the difference between public cloud and private cloud, their available solutions on the market and the way private cloud server enhances data security and privacy in cloud computing. The thesis is carried out in order to prove that building a private cloud server with end-to-end encryption not only enhances data security but also allows users to take the leading in securing their own confidential data.

The thesis was designed as a mix of comparative and experimental research that features articles, presentations, books, news, journals, and the project's result. Additionally, personal knowledge and experiences gained throughout the project are also discussed.

After a research had been carried out, the thesis discovered that more and more data and applications of users are moving to the cloud, mainly to the public cloud, which results in the rise of data security risks in public cloud. Therefore, public cloud and private cloud solutions are brought into comparison from the data security and privacy aspects.

The result of the thesis affirms the significant role of private cloud in securing users' data and privacy in cloud computing. Moreover, building a private cloud server enables users to have more control over their own hardware infrastructure, experience best speed in device synchronization, better privacy assurance, and lower in cost of cloud services. However, there are still various challenges in developing and maintaining the platform.

Keywords
Private Cloud, Data Security, Data Encryption, Cryptography, NextCloud, Client-side End-to-end Encryption

Table of contents

Abbreviations

Abbreviation	**Description**
3DES	Triple Data Encryption Standard
AES	Advanced Encryption Standard
API	Application Programming Interface
AWS	Amazon Web Services
CMAC	Cipher-based Message Authentication Code
CN	Canonical Name
CSP	Cloud Service Provider
CTR	Counter
DES	Data Encryption Standard
E2EE	End-to-end Encryption
ENCFS	Encrypted Filesystem for FUSE
FUSE	Filesystem in Userspace
GDPR	General Data Protection Regulation
GCM	Galois Counter Mode
GMAC	Galois Message Authentication Code
HMAC	Hash-based Message Authenication
HPC	High Performance Computing
IaaS	Infrastructure-as-a-Service
IEC	International Electrotechnical Commision
IETF	Internet Engineering Task Force
ISMS	Information Security Management System
ISO	International Organization for Standardization
MD5	Message Digest 5
NIST	National Institute of Standards and Technology

NSA	National Security Agency
OMAC	One-key Message Authentication Code
PaaS	Platform-as-a-Service
PBKDF2	Password-based Key Derivation Function 2
PHC	Password Hashing Competition
PHP	Hypertext Preprocessor
PHS	Password Hashing Scheme
QR	Quick Response
SaaS	Software-as-a-Service
SHA	Secure Hash Algorithm
SSE	Server-side Encryption
SSH	Secure Shell
SSL	Secure Sockets Layer
TOTP	Time-based One-time Password Algorithm
TDEA	Triple Data Encryption Algorithm
WebDAV	Web Distributed Authoring and Versioning
VM	Virtual Machine

1 INTRODUCTION

Cloud computing refers to a computing infrastructure and software model for enabling ubiquitous access to shared pools of configurable resources. These resources include computer networks, server, storages, applications and services, which can be rapidly configured and managed with minimal management effort over the Internet (Ngo 2017).

Cloud computing has moved beyond an interesting term and pervaded most of our daily lives. In fact, most people are using cloud services every day without noticing. In a report made in the U.S., 90% of global internet users are reported to be on the cloud and yet the consumer's awareness of cloud computing remains low. Half of the respondents answered that they either had not heard of the term cloud services or had not used them (Danova 2014.).

1.1 Thesis topic

Cloud users can access to the applications and data which located in the cloud from any location and at any time, data security and privacy issues in cloud computing are highly concerned. Especially in public cloud scenario, there are more and more security threats and challenges. The volume of cloud computing utilization, especially the utilization of public cloud, has been growing rapidly in the past few years, which results in the fact that a greater amount of sensitive data is potentially at risk. According to Heiser (2014), this is a transition period in cloud computing where the focus is shifting from the cloud service provider to the cloud user. The statement has brought up the question that if the main responsibility for securing data lies more with the cloud customer or the cloud service provider and how cloud users can enhance their own data security.

In Seattle, WA – 20 October 2017, the Cloud Security Alliance (CSA) released the list of top twelve data threats in cloud computing. The list comprises of critical threats which are data breaches, weak identity, insecure user interfaces (UIs) and application programming interfaces (APIs), system and application vulnerabilities, account hijacking, malicious insiders, advanced persistent threats, data loss, insufficient due diligence, abuse and nefarious use of cloud services, denial of service, and shared technology vulnerabilities. All data security threats cause negative impacts not only on the business but also on the customer. The worst cases of data threats are from public cloud servers. For example, BitDefender – an antivirus firm – was reported to experience a data breach in which customer usernames and passwords were visible in plaintext. The problem came from a security vulnerability in

BitDefender's public cloud application, which is hosted on Azure Web Services (AWS) (Goldman, 2015.).

Since there are various data security and privacy threats in cloud computing, data protection is not only essential but also obligatory in cloud servers. Many data protection techniques which are available in cloud computing such as: data encryption, access control, intrusion detection system, etc. (Jakimoski 2016, 49.). Among the available techniques, many IT security professionals consider data encryption to be one of the best ways to deliver data protection and client-side end-to-end encryption is the only way to keep your data secured in the cloud. According to Chang (2017), client-side encryption is an ultimate cyber-defense practice which has been referenced in the EU's GDPR (General Data Protection Regulation).

Most public open source cloud service providers only offer server-side encryption; however, client-side end-to-end encryption is said to be the more secure way to ensure that data is entirely protected. By 2014, ownCloud was reported to be the only open-source private cloud storage application that offered client-side end-to-end encryption and became popular among enterprises and home users (Salcedo 2014). In the recent years, there is a rise in the number of client-server software established that allows users to create their own file hosting services, manage their hardware infrastructures and have full control over the server. Some of these applications are open-source and they offer both client-side end-to-end encryption and server-side encryption, which help to secure data in the cloud. Some of the popular providers (besides ownCloud) are NextCloud, Sealife and Pydio (Ngo 2017).

1.2 Goals of this thesis

The research not only aims to give readers a deeper look into the concept of cloud computing and data security issues in cloud computing, but also provides them a private cloud solution for securing their data while enjoying the benefits of cloud computing.

In order to provide a solution for securing data in the cloud, the thesis will focus on comparing between data security solutions in public cloud and private cloud. The thesis will experiment the client-side end-to-end encryption in a private cloud server and aim to prove that end-to-end encryption in private cloud is the essential key to a prominent level of data security. The main purpose of the thesis is to provide a low-budget, integrated and secured cloud solution without the involvement of a third-party. The solution is best applied for private use (dedicated for home users).

1.3 Thesis tasks

The tasks of the thesis compose of two main parts, which are researching and implementing. The research stage consists of researching for general information on cloud computing, the latest issues in data security and privacy in cloud computing and the solutions for securing data in public cloud as well as private cloud. The research also provides reader with the information on data encryption in cloud computing (server-side encryption and client-side encryption) and current situations with public cloud service providers and private cloud storage applications.

The second stage of the thesis is implementing. The thesis's main project will be building a private cloud server with NextCloud. The implementing stage consists of the implementation plan, the working environment, the implementation phases, and testing and monitoring of the cloud solution. The main aim of this task is to prove that NextCloud's private cloud server is a highly integrated and secured private cloud solution.

1.4 Scope of this thesis

The thesis will provide detailed information on cloud computing and the latest issues in data security and privacy in cloud computing. General information on data security standards and data encryption algorithms will also be provided.

Client-side end-to-end encryption solution in public cloud server will be researched and stated in the theoretical background. Since the use of client-side encryption in public cloud may consist of a third-party software, which is a client-side encryption tool, the scope of this thesis will mainly focus on private cloud storage application. In order to provide an integrated cloud solution, client-side end-to-end encryption will be tested in a private cloud server built with NextCloud. Performance test and network traffic monitoring will be run on the prebuild private cloud server. The scope of the private cloud solution is for private use (home use) due to the physical computing capabilities of the Raspberry Pi.

1.5 Out of scope

The thesis does not include monitoring the network traffic on public cloud servers since the network will be under SSL security technology, which makes it impossible to monitor. Moreover, any attempt of breaking the encryption will not be carried out.

2 CLOUD COMPUTING

2.1 Introduction to cloud computing

The concept of Cloud Computing brings out different perceptions in different users. To most community and home users, cloud computing only refers to storing data and accessing software in the cloud and using associated services. To other, it is one of the most essential technologies for the existence of the internet world now and in the future (Deshmukh 2016). In fact, cloud computing has moved beyond an interesting term and pervaded most of our daily lives. However, the full potential and benefits of cloud computing cannot be reached without deep understanding its concept, architecture, models, capabilities, vulnerabilities, benefits and challenges (Ngo 2017).

Cloud computing is defined as a computing infrastructure and software model for enabling on-demand network, ubiquitous access to a shared pool of reliable and configurable resources (such as networks, servers, applications, storages and services), which can be rapidly provisioned with minimal management effort (Mell & Grance, 2009.).

2.2 The development of cloud computing

The development of cloud computing is a gradual evolution that first started back in the 1950s with mainframe computing. Back then, users were able to access the mainframe through dumb terminals, of which only function is to provide access to the central computer. After that, the concept of virtual machines was invented in 1970s. The virtual machine system took the mainframe computing one step further, enabled multiple virtual operating systems to be operated simultaneously in one single physical environment. The main technology that enables this evolution is called ‘virtualization’.

Not so long after, virtualized private network connections are developed and offered by telecommunications companies in the 90s. Virtualized private network connections helped telecommunications companies to reduce cost of building out multiple physical infrastructures and easily shift traffic if needed.

As the Internet became more popular and easier to access, virtualization was taken online. This whole evolution of how virtualization is utilized through the Internet gave birth to the concept of cloud computing.

2.3 **Characteristics**

According to Mell and Grance (2011, 2.), cloud computing model is composed of five essential characteristics, which are on-demand self-service, board network access, resource pooling, rapid elasticity and measured service.

The first characteristic to be explained is on-demand self-service, which means that the users can provision computing capabilities one-sidedly without interaction with service provider. These computing capabilities that can be provisioned are server time, network storage, etc. (Ngo 2017).

The second important characteristic of cloud computing is called broad network access. Users are provided with network end-points to be able to manage their cloud solutions. The most common end-points are standardized mechanisms such as the use of client- platforms which could be run on laptops, mobile phones or any other mobile devices (Ngo 2017).

The next key characteristic is resource pooling, which means the computing resources of the CSPs are pooled in order to serve multiple users using a model called multi-tenant model. There are different virtual resources as well as physical resources, which are dynamically assigned and reassigned to meet the requirements of customers (Ngo 2017.).

Rapid elasticity is the fourth characteristic of cloud computing to be mentioned. The cloud computing model's capabilities can be rapidly and flexibly provisioned and released depending on the users' needs. The capabilities available for provisioning usually appear to be unlimited and can be scaled up and down at any time by the customer (Ngo 2017.).

Last but not least, the last characteristic of cloud computing to be mentioned is measured service. Cloud computing systems can manage and optimize resource use by leveraging a metering capability automatically at some level of abstraction which is appropriate to the type of service (e.g. storage, bandwidth, etc.) Resource usage can be easily monitored, controlled and reported (Ngo 2017.).

2.4 **Architecture**

The architecture of cloud computing consists of components and subcomponents. These components normally include a front-end platform (thick client, thin client, tablets or mobile devices) and back-end platforms (infrastructure, servers, cloud storage, etc.) that are connected through a network (e.g. Intranet, Internet, etc.)

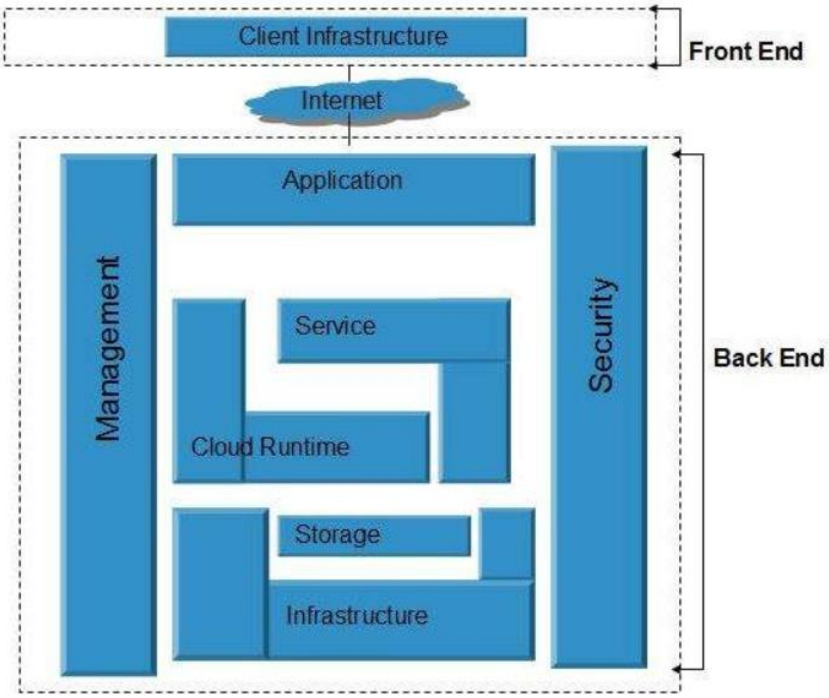


Figure 1. Cloud Computing's Generic Architecture (Tutorialspoint 2018)

In a cloud computing system, the client part is referred to as the front-end of the system. The front-end part consists of applications and interfaces which are used to access to the cloud platforms.

On the other hand, the backend refers to the resources that are required to run the cloud computing system and provide cloud services. The back-end consists of virtual machines, data storage, deployment models, servers and security mechanisms, etc. One of the main responsibilities of the cloud system's back-end is to offer built-in security mechanisms, network traffic control and protocols.

2.5 **Service models**

There are three major service models offered by cloud service providers which are Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) (Mell & Grance 2011, 2.).

Table 1. Shared Responsibility Model

Software as a Service (SaaS)	Platform as a Service (PaaS)	Infrastructure as a Service (IaaS)
User Access/Identity	User Access/Identity	User Access/Identity
Data	Data	Data
Application	Application	Application
Operating System	*Operating System*	Operating System
Virtualization	*Virtualization*	*Virtualization*
Network	*Network*	*Network*
Infrastructure	*Infrastructure*	*Infrastructure*
Physical Environment	*Physical Environment*	*Physical Environment*

The table above demonstrates the shared responsibility according to the cloud service models. The elements which have been underlined state the responsibilities of the users. On the other hand, the italicized elements belong to the cloud service providers' responsibilities.

Software as a Service (SaaS)

In Software as a Service model, the capability offered to the users is to use the provider's application software and databases running on a cloud infrastructure. The application software and databases can be Accessed from different client devices through thin client interface (e.g. web browser) or program interface. The CSPs hold entire management and control over the infrastructure and cloud platforms (e.g. cloud servers, operating systems, cloud storage, etc.), which are used to run the applications. The cloud users access the application software from the cloud clients and are not given any control over the infrastructure and platforms. The advantages of this model are the simplification in installation and maintenance, and minimization of the need for support. Some popular Software as a Service examples are: cloud-based Microsoft Office 365, Dropbox, Google Apps and Slack.

Platform as a Service (PaaS)

In Platform as a Service model, users are able to deploy the applications (user-acquired or user-created) onto the cloud infrastructure. The applications created or acquired by cloud's users can be established by using programming languages, libraries, tools and services which are supported by the CSP. The control and management of the cloud infrastructure (servers, cloud storage, network and operating system) belong to the CSP. However, the cloud users do have control over the deployed applications and the configuration settings for the application-hosting environment. This advantage of this model is to help user cut down the cost and the complexity of purchasing and handling the hardware infrastructure and software layers. Some popular Platform as a Service providers are: Google App Engine, Heroku, Amazon AWS, and Windows Azure Cloud Services. In addition, there are many specialized PaaS such as BaaS (Blockchain as a Service), iPaaS (Integration Platform as a Service), and dPaaS (Data Platform as a Service).

Infrastructure as a Service (IaaS)

In Infrastructure as a Service model, users are able to provision processing, cloud storage, networks and other computing resources where the users deploy and run arbitrary software, which include operating systems and applications. The control and management of the cloud infrastructure belong to the CSPs. Nevertheless, the users are capable of managing and controlling the operating systems, cloud storage and the deployed applications. Additionally, users can have limited control over specific networking components (e.g. firewalls).

2.6 **Deployment models**

There are four main deployment models in cloud computing which are public cloud, private cloud, community cloud and hybrid cloud (Mell & Grance 2011, 2.).

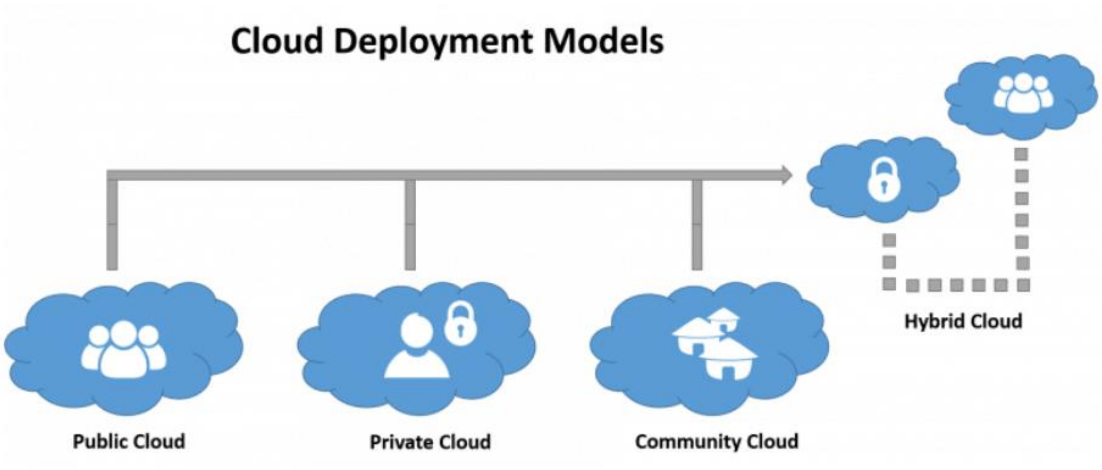


Figure 2. Cloud computing deployment’s model (Fu, A. 2017)

Public cloud

Firstly, a cloud server is categorized as a public cloud when the services rendered over a network which is open for public use. The cloud service provider makes resources (applications, storage or virtual machines) available for public use over the Internet. Public cloud service may be free of charge or offered on a pay-per-usage model (Rouse, M. 2017b)

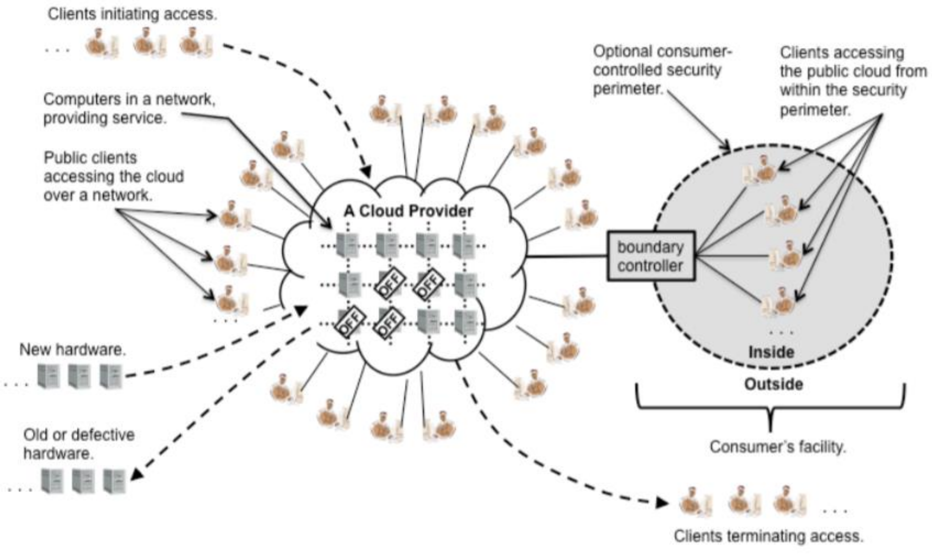

Clients initiating access.
Computers in a network, providing service.
Public clients accessing the cloud over a network.
New hardware.
Old or defective hardware.
A Cloud Provider
OFF
OFF
OFF
OFF
boundary controller
Optional consumer-controlled security perimeter.
Clients accessing the public cloud from within the security perimeter.
Inside
Outside
Consumer's facility.
Clients terminating access.

Figure 3. Public Cloud Scenario (Badger, Grance, Patt-Corner & Voas 2012, 4-13)

Within a public cloud scenario, the cloud provider's storage resources are normally large, and the connecting links are typically implemented over the Internet. In addition, the cloud provider usually serves multiple different cloud clients.

Public cloud platforms are the most popular cloud computing service among general users. Public cloud computing has its resources, application storage offered to the consumers over the internet. Most of the cloud-based applications such as SaaS offerings (online cloud-based applications, cloud storage…) utilizes public cloud computing platform. Public cloud offers a lot of benefits to its users such as: scalability, cost-effectiveness, reliability and location independent.

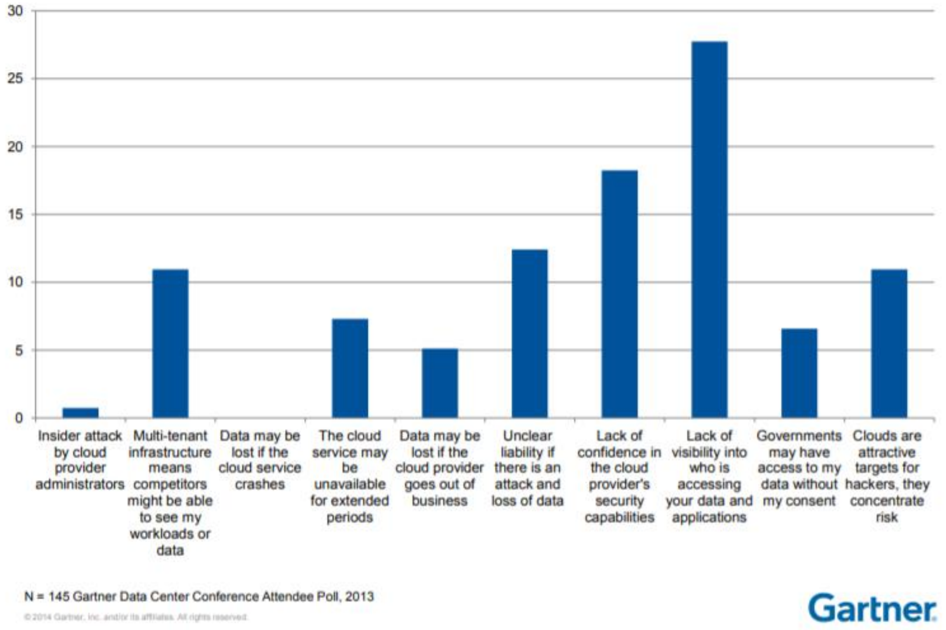
One main benefit that public cloud computing offers is great scalability. Public cloud storages are often scalable in an effortless way. The increase and decrease storage, adding and removing software are flexible on demand. In the same manner, public clouds are cost-effective since they offer pay-as-you-go approach. When it comes to storage, even a limited amount of storage is offered to users without any up-front costs. For example, Dropbox offers the first 2GB, iCloud and OneDrive offer 5GB and Google Drive offer 15GB of storage with no cost. The basic upgrade to 1TB in Dropbox, iCloud (automatically to 2TB) and Google Drive will cost users approximately ten euros per month.

Public clouds can be much reliable. When it comes to the use of public cloud, users do not need to take care of maintenance and update of the cloud-based applications offered by the cloud service providers. Moreover, public cloud platform offers reliability in which no single point of upgrading, maintaining and failure will interrupt the use of services. Lastly, services (e.g. SaaS) on public cloud platforms can be accessed easily over any devices that are connected to the Internet at anytime and anyplace.

As public cloud offers various appealing benefits, there are a few disadvantages that comes along. The major downsides of public cloud computing are higher data security and privacy risk, performance and flexibility. Users of public cloud services usually have limited visibility and less or no control over data security and privacy since the services and infrastructure are provisioned from the cloud service provider’s data center. Users are not offered a guaranteed way to control and authorize access to their data in public cloud. Shared resources between multiple public cloud tenants bring up more vulnerabilities making security risks more intense. Compared to private cloud, network performance is less reliable in public cloud. Moreover, public cloud platforms are less flexible since they limit the customization of services and resources.

In the Gartner Data Center Conference (Heiser, 21 August 2014), it has been recorded that there were 10 primary issues with security and privacy in public cloud, which stated below.

Figure 4. Issues with Security and Privacy in Public Cloud (Heiser, 21 August 2014).



N = 145 Gartner Data Center Conference Attendee Poll, 2013

© 2014 Gartner, Inc. and/or its affiliates. All rights reserved.

The ten security and privacy risks in public cloud computing mentioned above shows that users have lack of confidence and visibility when it comes to data security in public cloud.

Private cloud

Secondly, a cloud server is called private cloud when the cloud infrastructure operated for a single organization, managed internally or by a third-party and hosted internally or externally. The private cloud is created, maintained and managed by a single organization. Moreover, the private cloud infrastructure and resources might be available in the organization's data center (on-premises) or in a separate infrastructure (off-premises) (Rouse, M. 2017c).

Private clouds are more popular among large organizations and enterprises. Private cloud refers to an infrastructure that is dedicated to only one single organization. A private cloud server can be managed internally (on-site private cloud) or by a third-party (outsourced private cloud). Private cloud has many advantages regarding data security such as: strong security from external threats, more control over the cloud server, higher reliability (Badger, Grance, Patt-Corner & Voas 2012, 4-4).

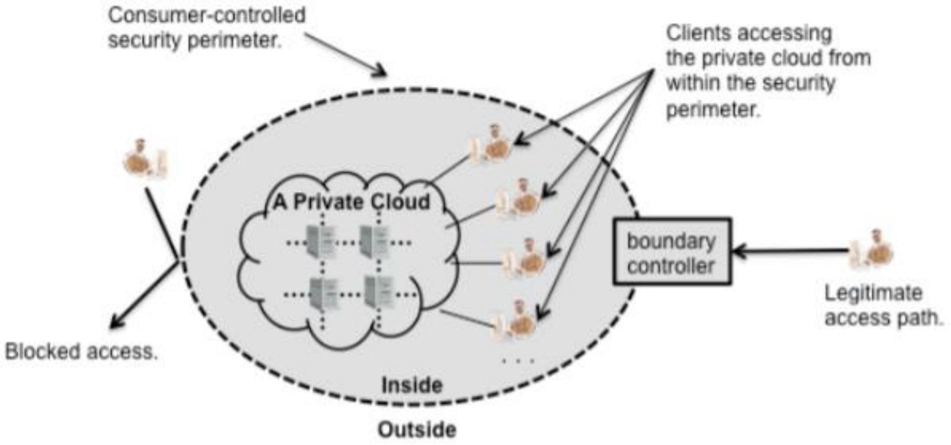
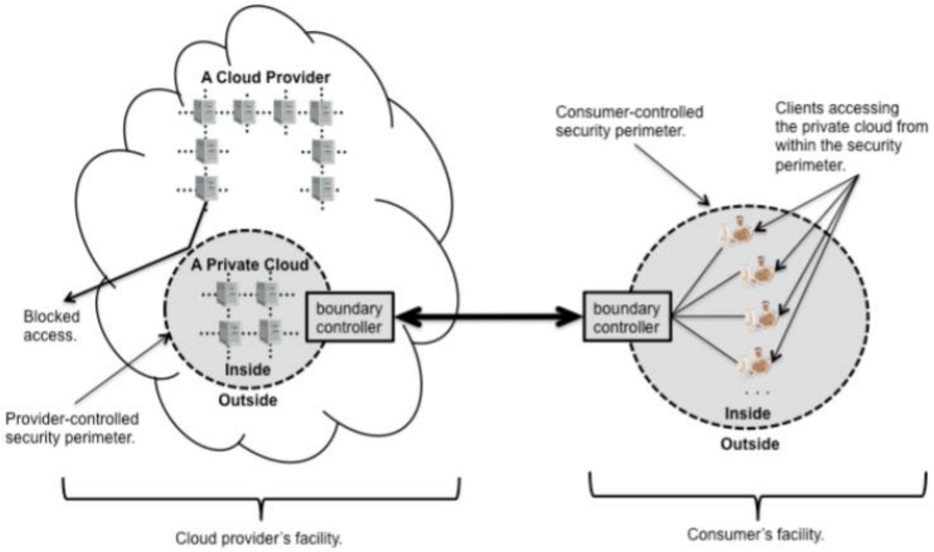


Figure 5. On-site Private Cloud Scenario (Badger, Grance, Patt-Corner & Voas 2012, 4-4)

In case of an on-site private cloud scenario, the security bounder covers both the resources of the private cloud server as well as the cloud users. The security perimeter belongs to the user's responsibility whether or not to implement it and the cloud clients can access the private cloud server within the security bounder. If the security perimeter is set and controlled, only specified authorized access is allowed through the boundary controller. The on-site private cloud scenario potentially offers prominent level security from external threats since the user can decide to implement a proper strong security perimeter to the secure the private cloud server. However, the biggest disadvantages of a typical on-site private cloud server are high up-front costs and limitations from performance, data import and export.

Figure 6. Out-sourced Private Cloud Scenario (Badger, Grance, Patt-Corner & Voas 2012,


A Cloud Provider
A Private Cloud
boundary controller
Inside
Outside
Blocked access.
Provider-controlled security perimeter.
Consumer-controlled security perimeter.
Clients accessing the private cloud from within the security perimeter.
boundary controller
Inside
Outside
Cloud provider's facility.
Consumer's facility.

4-4)

In case of an out-sourced private cloud scenario, there exist two different security perimeters, one implemented and controlled by the cloud provider and the other is implemented by the cloud user. The two perimeters are connected by a secured communicating link. The level of data security in an out-sourced private cloud server depends on both security perimeters as well as the connecting link between them. The out-sourced private cloud, as similar to the on-site one, embraces strong security from threats. However, the main different is that the security techniques have to be applied to both security perimeters and the connecting link also have to be secured.

Private cloud server is built dedicating to one single organization. The infrastructure and services are implemented and provisioned in an on-site data center (regarding on-side private cloud) or in a third-party data center (regarding out-sourced private cloud), giving users more control over the security perimeters as well as the data location and data itself. Therefore, data security and privacy's levels are higher, and the risk of multitenancy is lower than in public cloud. Not only giving users high-level of data privacy, private cloud hands users more flexibility and control over the cloud server. Users can take lead in controlling and monitoring customization over the cloud infrastructure, server management, authorization and data.

Nevertheless, private cloud has lots of disadvantages such as up-front cost (hardware and equipment) and operating cost (maintenance and upgrade). For instance, ownCloud offers solutions for enterprise subscription which cost up to seven thousand euros for fifty users and more than eleven thousand euros for one hundred users (ownCloud, 2018.). Moreover, remote access from mobile users to the cloud is a drawback due to high security level.

Community cloud

Cloud infrastructure in community cloud is provisioned for exclusive use by a specific group of users from organizations that share the same interests (such as mission, requirements, compliance consideration…). Community cloud can be managed by one or more organizations in the community or a third party (or a combination of them). Similar to private cloud, community cloud server may be hosted internally or externally. The community cloud hosted internally is called on-site community cloud, and the one which is hosted externally is called out-sourced community cloud.

Hybrid cloud

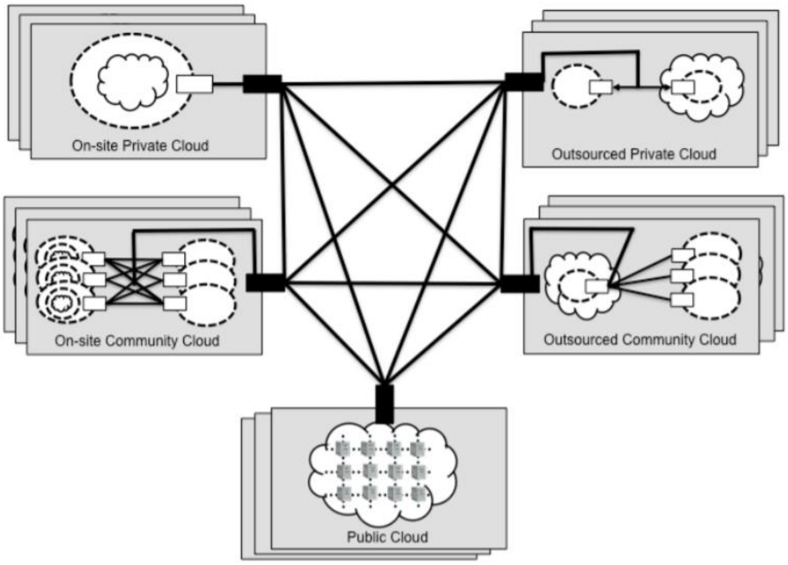


Figure 7. Hybrid Cloud Scenario (Badger, Grance, Patt-Corner & Voas 2012, 4-4)

Fourthly, a cloud server is called hybrid cloud when the cloud infrastructure is a combination of two or more different cloud infrastructures (public, private or community). The distinct cloud infrastructures to be combined must remain unique entities. However, the infrastructures must be bound together by proprietary technology which enables portability of application and data.

Others

Beside the four main deployment models of cloud computing, there are other deployment models such as distributed cloud, multi-cloud and HPC (high-performance computing) cloud. A distributed cloud refers to a cloud platform which can be assembled from a distributed set of machines from distinct locations, connected to a hub service or single network. Multi-cloud refers to multiple cloud computing services in a single heterogeneous architecture (Rouse, M. 2017a). HPC cloud refers to utilization of the cloud infrastructure and services to execute high performance computing applications (Netto, Calheiros, Rodrigues, Cunha & Buyya 2018, 1.).

3 DATA SECURITY IN CLOUD COMPUTING

3.1 Introduction to data security

Data security and privacy in cloud computing has not only been a popular topic between IT professionals, enterprises and cloud users, but also one of the biggest challenges of cloud computing. Data security and privacy is stated to be the top issue to be considered before adopting cloud system. The main reason is traded back to the main technology that enables cloud computing which is virtualization (Hamdaqa & Tahvildari 2012, 72.). Virtualization alters the relationship between the hardware and the operating system, represents the computing system, storage and networking itself which must be properly configured and secured. The use of virtualization in cloud computing infrastructure brings data security concerns for the public cloud service's users (Winkler 2011).

Cloud computing security is defined as a broad set of standardized policies and technologies to protect data, applications and the infrastructure of cloud computing.

3.2 Information security standards

Security standards are the techniques which are set in published materials. The ISO27K suite provides more than fifty published standards. Especially, from ISO/IEC 27001 to ISO/IEC 27006, the standards which specify a formal information security management system are stated (ISO 27001 Security 2018). Moreover, ISO27K is stated to help with achieving GDPR compliance (Dutton, 2017.).

The EU General Data Protection Regulation (EU GDPR) refers to a regulation in the European Union law on data protection and privacy dedicated to all individuals within the EU. The regulation is adopted in 2016 and will become enforceable on 25 May 2018. The EU GDPR encourages the use of ISO 27001 certification scheme in order to show that the organization is managing its data security actively and in line with international best practices (Dutton, 2017.).

3.2.1 ISO/IEC 27001 to ISO/IEC 27006

An information security management system (so-called ISMS) is a set of policies and procedures concerning the management of information security risks.

The ISO/IEC 27001:2013 provides the security techniques and requirements to formally specify an Information Security Management System. The information security management system is an management framework which the organization can utilize to identify, analyze and address its information security risks. The fifteen essential documentation and requirements for the standardized certification are: the information security management system scope (clause 4.3), the information security policy (clause 5.2), the process of information risk assessement and treatment (clause 6.1.2 and 6.1.3), the objectives of information security (clause 6.2), proof of competence of the people working in information security management (clause 7.2), ISMS-related documents (clause 7.5.1b) ,documents of operational planning and controls (clause 8.2), results of risk assessments and decisions regarding risk treatment (clause 8.2 and clause 8.3), proof of monitorning, measurement and analyzing of information security (clause 9.1), the ISMS internal audit program as well as the results of audits conducted (clause 9.2), proof of the ISMS's management reviews (clause 9.3), proof of non-conformities identified and corrective actions arising (clause 10.1), and many others.

ISO/IEC 27002:2013 refers to a code of practice – an internationally standard of good practice for information security management. The ISO/IEC 27001 standard utilizes ISO/IEC 27002 in order to specify the best practices of information security controls within the information security management system.

ISO/IEC 27003:2017 provides the detailed and pragmatic explanations and guidance on the implementation of ISO27k standards, especially ISO/IEC 27001:2013.

ISO/IEC 27004:2016 refers to the security techniques concerning monitorning, measurement, analysis and evalution issues of the information security management system. This standard expands the ISO/IEC 27001:2013 on clause 9.1 – which has been mentioned above.

ISO/IEC 27005:2011 supports the general concepts of an ISMS in ISO/IEC 27001:2013 by providing the guidelines and security techniques concerning information security risk management.

ISO/IEC 27006:2015 is published as an accreditation standard, which comprises of requirements for certification bodies on the formal procedure to follow when auditing the client's ISMS(s) according to ISO/IEC 27001:2013 (ISO 27001 Security 2018).

3.2.2 **Other standards**

Furthermore, there are various cloud security standards initiatives published (such as ISO/IEC 27017 and ISO/IEC 27018) that provide specifically detailed guidance and recommendations for cloud service providers (CSPs) and cloud service users. The ISO/IEC 27017:2015 provides guidelines for information security controls, which based on ISO/IEC 27002 for cloud computing services. The ISO/IEC 27018:2014 provides guidelines for protection and security of Personally Identifiable Information in public clouds. They are both built upon the basis of ISO/IEC 27002 and expands ISO/IEC 27001 in details (ISO 27001 Security 2018).

Moreover, there are also general IT security standards (for example ISO/IEC 38500 and X.509 certificates) which can be applicable to cloud environments. Cloud service customers should be aware of these standards and make sure that the CSPs support them (Cloud Standards Customer Council, 2016.).

3.3 **Security issues in cloud computing**

From the cloud providers' perspective, there are many open concerns in information security, which are: risk of unintended data disclosure, data privacy, system integrity, multi-tenancy, browsers, hardware support for trust and key management.

Typically, users tend to store non-sensitive and sensitive data in different directories on a cloud system. By doing so, sensitive data is expected to be handled in a more secure way to avoid the risk of unintended sensitive data distribution. However, if a user wishes to use cloud platform mostly for non-sensitive computing, while retaining the prominent level of security for sensitive computing, care must be taken so that sensitive data will be stored in encrypted form.

Secondly, protecting data privacy in any cloud computing system (or any other computing system) is not only a technical challenge, but also an ethical and legal concern. Especially in cloud computing, of which nature is distributed system, users have less awareness over where their data is stored physically and who can have the access to their data.

Thirdly, system integrity is one of the main issues in every cloud system. Within a cloud, there are separate groups such as providers, administrators and users. The main challenge is being able to partition access rights to each of those stakeholders, while preventing malicious attack.

Fourthly, the nature of cloud computing is sharing the resources on the cloud service provider’s side. For SaaS clouds, different users may share the same cloud-based application or storage; for IaaS clouds, different virtual machines (VMs) may share the same hardware via a hypervisor… Since all the sharing mechanisms happen at the provider’s facility entirely depend on complex utilities to keep user workloads isolated, the risk of data isolation failure exists. The challenge is to build proper and secured workloads logical separation.

The next data security risk in cloud computing comes from the browsers. It has been reported that browsers were vulnerable and harboured security flaws in nearly every security challenge. If a user’s browser is destabilized, all of the data that user entrusted to the cloud provider will be at risk. The important challenge is to build confidence that browsers are not subverted by restricting browsers’ types and limiting plug-ins.

In some cases, hardware support can deliver the trustworthiness of remote systems to the users. The Trusted Platform Module (TPM) was developed with that purpose. However, it was reported to have a weak point in its trust chain when virtual machine migrated. Many groups have been making effort in virtualizing the trusted platform module (TPM), or to establish an argument in which a migrated virtual machine can re-establish trust on different hardware. However, the issue remains.

Last but not least, cryptography key management is a critical issue in the cloud. The issue is that zeroing a memory buffer may or may not delete a key if the memory was backed by a hypervisor which makes the memory persistent; or the virtual machine is having a snapshot for recovery… (Badger, Grance, Patt-Corner & Voas 2012, 8-7.).

3.3.1 **Top threats in security**

In Seattle, WA – 20 October 2017, the Cloud Security Alliance released the top twelve critical threats to data security in cloud computing. The top threats which are published in the report are data breaches, weak identity, insecure APIs, system and application vulnerabilities, account hijacking, malicious insiders, advanced persistent threats, data loss, insufficient due diligence, abuse and nefarious use of cloud services, denial of service, and shared technology vulnerabilities.

A data breach (unintended data disclosure) refers to the situation is which sensitive, confidential and secured data is released or stolen by an unauthorized individual. These confidential data can be referred to financial information, health information, personally identifiable information, intellectual property or trade secrets. The risk of data breaching is always a top concern for cloud service providers as well as users. In the end of 2013, Adobe had a terrible data breach (so-called data leakage) case. An estimated number of thirty-eight million customer records (including debit and credit cards' information) was out of control. In 2015, BitDefender – an antivirus firm – experienced a data breach case where their customer usernames and passwords stolen. In fact, there are lots more data breach cases that resulted in loss of customer's data as well as financial loss for the company.

Secondly, insufficient identity, credential and access management can be the cause of data breaches and enabling of attacks. It refers to the lack of multifactor authentication, weak password, scalable identity access management system and automated rotation of passwords, certificates and cryptographic keys. It results in enabling unauthorized access to data and damaging the organizations and end user's confidential information.

Thirdly, software user interfaces (UIs) and application programming interfaces (APIs) are normally the most exposed part of the system. These interfaces must be strengthened in order to protect against malicious attempt (or accidental attempt) circumventing policy since they are the only assets with IP addresses available outside the secured boundary. In the middle of 2015, the United State Internal Revenue Service (US IRS) exposed approximately 300,000 records vis an insecure API.

System vulnerabilities can be referred to exploitable bugs in programs which attackers can utilize to infiltrate a system so that they can steal data, take control of the computer system or disrupt the service operations. In 2014, Bash's Shellshock bug was reported to have various successful attacks.

Account hijacking can be explained as an attack where the attacker uses the stolen credentials to access critical areas of cloud computing services, manipulate data, return fake information and put the confidentiality, availability and integrity of data at risks.

Malicious insiders are the former/current employees or partners who had/have the authorized access to the organization's system, network and data and intentionally or unintentionally use that access in a way that negatively affect the integrity and confidentiality of the information and the systems.

Next, advanced persistent threats (APTs) are a type of cyber-attack in which the attacker establishes a foothold in the targeted companies' computing infrastructures. After that, the attacker will smuggle the companies' intellectual property and confidential data.

Data loss explains the situation in which the data stored in cloud can be lost for reasons (not by attackers). Data loss can be caused by accidental deletion by the cloud providers (CSPs), loss of encryption key on the client-side, etc.

Insufficient due diligence refers to the rush of adopting cloud technologies and cloud service providers without proper due diligence. An organization is exposed to various financial, commercial, technical, legal and compliance risks that potentially threaten its success.
The next threat to be mentioned is abuse and nefarious use of cloud services. It refers to the free cloud service trials, insecure cloud service deployments or fraudulent account sign-ups through payment instrument fraud. All of the above will result in the cloud computing's service models (SaaS, PaaS and IaaS) being exposed to malicious attacks.

Denial of Service attacks (or DoS) are the type of attack in which the attacker aims to prevent users from accessing their data or applications by slowing down the entire system. The attacker does this by forcing the cloud service to consume a huge amount of system resources (processor power, disk space, memory or network bandwidth).

Finally, shared technology vulnerabilities refer to vulnerabilities caused by sharing the same infrastructures, platforms or applications. The underlying components of the infrastructure may not offer strong isolation properties for a multi-tenant architecture (regarding IaaS), redeployable platforms (regarding PaaS) or multi-customer application (regarding SaaS).

Among the twelve top threats mentioned and explained above, encryption and key management is referenced by CSA Security Guideline in order to prevent seven data security risks (which are data breaches, insufficient identity, credential and access management, insecure interfaces and APIs, account hijacking, malicious insiders, insufficient due diligence, and shared technology vulnerabilities) (Cloud Security Alliance, 2017.). Therefore, the utilization of proper encryption and key management techniques are essential to cloud security and it can result in a highly secured information system. According to Sen and Tiwari (2017, 70.), the most significant solution for users is data encryption.

4 DATA ENCRYPTION IN CLOUD COMPUTING

4.1 Cryptographic algorithms

Cryptographic services are provided for the purpose of sensitive data protection. In cryptography, data encryption in the cloud is the process of encoding the data or information in a way that only authorized users have the access right. Data can be encrypted before entering the cloud using symmetric-key or asymmetric-key. In symmetric-key scheme, there is only one key for encryption and decryption. On the other hand, in asymmetric-key schemes, there are two different keys. The encryption key is published for users to encrypt data and the decryption key is held by the CSP.

In cryptography, there are many important cryptographic algorithms and functions. These algorithms and functions are symmetric-key algorithms (private-key algorithms), asymmetric-key algorithms (public-key algorithms), block-cipher mode of operation, cryptographic hash functions, message authentication code, and key derivation functions.

4.1.1 Symmetric-key algorithms

Symmetric-key algorithms are cryptographic algorithms that use the same keys for both encryption and decryption. The data before encryption is called plaintext and the one after encryption is call ciphertext.

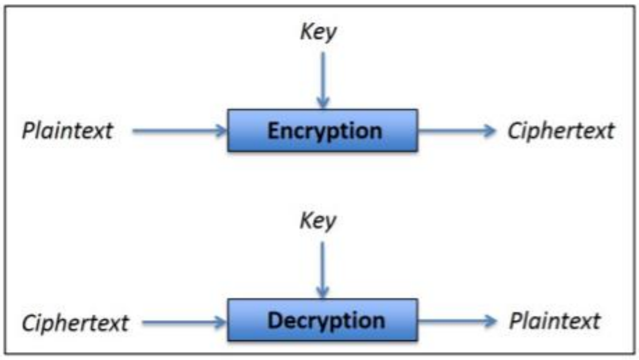


Figure 9. Encryption and Decryption using Symmetric-key algorithm (Barker 2016, 19.).

In symmetric-key algorithms, there are two main types which are block ciphers and stream ciphers. Stream ciphers handle encryption in the way that they encrypt the plaintext digits (normally bytes), or letter of a message one at a time, using the corresponding digit of the keystream.

On the other hand, block ciphers handle encryption in the way that they take a number of bits and encrypt them as a single unit, give plaintext a specific padding so that it is a multiple of the block size. There are many approved standards and algorithms that have been approved by NIST under block-cipher algorithms, which are: DES, 3DES (TDEA), AES, etc. AES is believed to be one of the most secure algorithms for data encryption in cloud computing.

DES algorithm

DES stands for Data Encryption Standard and is a symmetric-key algorithm that was developed in the early seventies by IBM. The algorithm works by taking a string in plaintext and putting through several complicated operations called Feistel functions to achieve the cipher text of the input plaintext string. DES has a 64-bit block size and uses a key of the same length for its encryption process. It is also important to note that the key only uses 56 bits of the available 64 bits. This is done to leave 8 bits for only checking parity operations which will thereafter be discarded (NIST, 2018a).

TDEA (3DES) algorithm

Triple DES (3DES) or so-called Triple Data Encryption Algorithm (TDEA) is the consequently follower of the DES algorithm. It therefore makes improvements to the ciphers key size of 56-bit which were not enough anymore to avoid brute force attacks. Triple DES, as the name suggests, uses three of the Data Security Standard 56-bit keys for its encryption process, and makes the encryption process secure again whilst avoiding the need to come up with a completely new block cipher algorithm (NIST, 2018a.).

AES algorithm

The Advanced Encryption Standard (AES) is listed as one of the most popular and secure encryption algorithms available. AES is not only publicly accessible but also the cipher which the National Security Agency (NSA) uses for protecting their top confidential documents.

The concept of Advanced Encryption Standard was started in a competition (organized by the National Institute of Standards and Technology in 1997) searching for a potential replacement of Data Encryption Standard (DES). AES (with the original name Rijndael), developed by two Belgian cryptographists, came on top of several other competitors due to its excellence in security, flexibility and performance.

AES algorithm is based on multiple permutations, substitutions and linear transformations (each operation is executed on data blocks of 16 byte). These operations are repeated in several “rounds” (starting with the Initial Round, continuing with Rounds and ending with Final Round). During each round, only one unique RoundKey will be calculated out of the encryption key and then incorporated in the calculations.

The main advantage of block cipher over stream cipher is that the change of a single bit (either in the plaintext or in the key) will result in a completely different ciphertext block. Compared to the 56-bit key of DES, AES offers three different key’s lengths, which are 128-bit key (AES-128), 192-bit key (AES-192) and 256-bit key (AES-256) (NIST 2001). At the present time, there has been no practical attack exists that could break the AES encryption without knowing the key to decrypt the data when the algorithm is properly implemented.

4.1.2 Asymmetric-key algorithms

Asymmetric-key algorithms are also used for data encryption, but these algorithms are slower compared to symmetric algorithms. Public-key algorithms are not normally used for general data encryption, however, they can be used for key management. One asymmetric-key algorithm can be used in data encryption in cloud computing is RSA. The RSA algorithm is utilized especially for verification of digital signatures (Barker 2016, 19.).

RSA algorithm

RSA is reported to be one of the most popular and successful asymmetric encryption systems nowadays for securing data in transmission. RSA was originally developed by Clifford Cocks (who worked for a British intelligence agency called GCHQ), however, it was not published since it was classified as top-secret. After that, the RSA algorithm was re-discovered in 1977 by Rivest, Shamir and Adleman (R-S-A).

RSA encryption system works based on two different keys: one public key and one private key. A message or a folder which is encrypted by one of the key can only be decrypted by the other one. Since the private key is not related and cannot be calculated out from the knowledge of the public key, the latter is available to public.

RSA is widely known for the use of digital signatures. When a document is signed, a fingerprint encrypted utilizing RSA system is attached to the file. This process enables the receiver to verify the identity of the sender as well as the integrity of the file. The RSA is based on mathematical issue of integer factorization. The to-be-encrypted message is treated as a large number. When the message is being encrypted, it is raised to the power of the encryption key, then divided with the remainder by a fixed product of two primes. After being successfully encrypted, the plaintext can be retrieved again by repeating the process with the other encryption key. It has been recorded that a 768-bit key (RSA-768) has been broken. Therefore, modern cryptosystems use 3072-bit key as the minimum (Boxcryptor, 2018.).

4.1.3 Block-cipher mode of operation

In cryptography, a block-cipher mode of operation refers to an algorithm which uses a block cipher in order to provide an information service (e.g. authenticity or confidentiality). There are modern block-cipher modes of operation that effectively combine authenticity and confidentially, which can be referred as authenticated encryption modes (NIST, 2018a).

A block cipher itself is only suitable for the process of encryption and decryption of one fixed-length group of bits (a block). A block cipher mode of operation's main function is to describe how to repeatedly apply a block cipher operation to securely transform amounts of data larger than a block.

Most of the modes of operation require an initialization vector (a unique binary sequence) for each encryption operation. The initialization vector must be random as well as non-repeating. The vector is utilized to ensure different ciphertexts are generated even when the same plaintext is encrypted several times independently with the same cryptographic key (Huang, Chiu & Shen, 2013).

Block cipher modes of operation normally operate on the whole block and the modes require that the last part of data must be padded to a full block in case the data size is smaller than the current block size (Huang, Chiu & Shen, 2013.). However, there are block-cipher modes that do not require padding (NoPadding) since they can treat the block cipher as a stream cipher effectively.

The common modes of operation are ECB (Electronic Codebook), CBC (Cipher Block Chaining, CFB (Cipher Feedback), OFB (Output Feedback), and CTR (Counter). Especially, a number of modes are specifically designed for the use of authenticated encryption, which are CCM (Counter with Cipher-block Chaining Message Authentication Code) and GCM (Galois Counter Mode).

Moreover, the GCM (Galois Counter Mode) is the block cipher mode of operation, which has been defined for symmetric key cryptographic block ciphers with a block size of 128 bits. GCM utilizes universal hashing over a binary Galois field in order to provide authenticated encryption. The Galois Hash is utilized for the authentication and the AES block cipher is used for encryption in CTR mode (counter mode) of operation. AES/GCM Authenticated Encryption is designed for high performance and has been proved to be the best performing Authenticated Encryption combination among NIST standard options. (Gueron, 2013).

4.1.4 **Cryptographic hash functions**

A cryptographic hash function is defined as a special class of hash function. The cryptographic hash function contains certain properties that make it suitable to be used in cryptography. Its main purpose is to map data of arbitrary size to a fixed-length hash value, and it is designed to be a one-way function. The input data into the cryptographic hash functions is called a message and the output is called a message digest (or digest).

The popular hash functions are MD5, SHA-1, SHA-2, and SHA3. The applications of cryptographic hash functions are file verification, password hashing, proof-of-work system, file/data identifier, pseudorandom generation, and key derivation.

MD5 algorithm

The MD5 (so called Message Digest 5) algorithm was designed to be cryptographic hash function which is used to produce a 128-bit hash value. However, MD5 has been found to suffer from extensive vulnerabilities (Joux 2004, 306.).

SHA algorithms

The Secure Hash Algorithms (SHA) refer to cryptographic hash functions which are published by NIST as a U.S Federal Information Processing Standard. The Secure Hash Algorithms includes SHA-0, SHA-1, SHA-2, and SHA-3.

SHA-0 was first published under the name “SHA” in 1993. However, the hash function has been reported to occur many collisions, which leaded to the development of SHA-1 (Joux 2004, 306.).

SHA-1, so called Secure Hash Algorithm 1, refers to the cryptographic hash function which resembles MD5. SHA-1 was first designed by the NSA (National Security Agency) to serve as a part of the DSA (Digital Signature Algorithm). The main functions of SHA-1 is taking an input and producing a 160-bit hash value (NIST, 2018b).

SHA-2 refers to a set of cryptographic hash functions which are designed by the NSA in 2001. The SHA-2 hash functions are built based on the Merkle-Damgard structure and consist of fix different hash functions. The hash functions in SHA-2 are SHA-224, SHA-256. SHA-384, SHA-512, SHA-512/224, and SHA-512/256. These hash functions are used to produce hash values which are 224, 256, 384 or 512 bits (NIST, 2018b).

SHA-3 was designed differently from MD5, SHA-1 and SHA-2. However, SHA-3 is still a member of the SHA family of standards. SHA-3 was published by NIST in 2015 in order to directly substitute SHA-2. SHA-3 uses a construction called sponge construction. In cryptography, sponge construction consists of algorithms which take input bit stream of any length and provide the output of any desired length. The SHA-3 hash functions consist of SHA3-224, SHA3-256, SHA3-384, and SHA3-512. Moreover, there are two derived functions of SHA-3, which are the extendable-output functions. The two extendable-output functions are SHAKE128 and SHAKE-256 (NIST, 2018b).

4.1.5 **Message authentication codes**

A message authentication code (or MAC) refers to a short piece of information which is used to authenticate a message. To put in other words, MAC is used in order to confirm that the message is sent from the specific sender and has not been changed from its original form. The main purpose of message authentication code is to protect the integrity as well as the authenticity of the message (Bellare, Canetti and Krawczyk, 1996.).

Message authentication code algorithms can be constructed from cryptographic primitives (e.g. block-cipher algorithm or cryptographic hash functions). Some popular MAC functions are GMAC (Galois Message Authentication Code), HMAC (Hash-based Message Authentication Code), and OMAC/CMAC (One-key Message Authentication Code).

GMAC

Galois Message Authentication Code (GMAC) can be utilized as an incremental message authentication code which is only dedicated for the authenticity of the message. GMAC is considered as a variant of GCM (Galois-Counter Mode).

OMAC/CMAC

One-key Message Authentication Code (OMAC) refers to a block-cipher based message authentication code algorithm. There are two official OMAC algorithms which are OMAC1 and OMAC2. OMAC1 is considered to be equivalent to CMAC (Cipher-based Message Authentication Code).

HMAC

Hash-based MAC, so called keyed-bash MAC, involves a cryptographic hash function and a cryptographic key. HMAC can be utilized to verify both data authentication and data integrity of a message. More specifically, HMAC is used to determine whether or not a message, which is sent over an insecure channel, has been changed since its original, provided that both sender and receive share a secret cryptographic key.

At the beginning of the process, the sender will compute the hash value for the original message and send the message and the hash value (as a single message). The receiver will recalculate the hash value on the message with the secret key and check if the HMAC code generated by the receiver matches the one received from the sender. Any change to the message or on the hash value itself will result in a mismatch. Therefore, if the original message and hash values match, the message's authenticity and integrity are proved (Microsoft 2018).

Cryptographic hash functions can be used in the process of calculating HMAC. The hash functions frequently used are MD5 and SHA-1. The cryptographic strength of Hash-based MAC depends on the strength of hash function (size of hash output and quality and size of the key).

HMAC uses two passes of hash computation. Firstly, the secret cryptographic key is used to derive two keys (inner-key and outer-key). After the first pass, an internal hash result derived from the message and the inner key are generated. After the second pass, the final HMAC output hash is derived from the inner hash result and the outer-key. The final output hash is 160 bits length.

HMAC-MD5 was stated not to be exposed to a practical vulnerability when being used as a MAC even though MD5 itself has been found to suffer from extensive vulnerabilities. However, it was also added that HMAC-MD5 should not be used for a new protocol design (Turner & Chen 2011, 1.).

HMAC-SHA1 is constructed from the SHA-1 hash function and used as a Hash-based Message Authentication Code.

4.1.6 Key derivation functions

In cryptography, key derivation functions utilize pseudorandom function (which can be used to emulate a random oracle) in order to derive one or more secret cryptographic keys from a secret value (e.g. master key, password or passphrase). The cryptographic hash functions (e.g. SHA-1, SHA-2, etc.) are used as pseudorandom functions for key derivation (Camenisch, Fischer-Hubner & Rannenberg 2011, 185.).

Some common purposes of key derivation functions are password hashing, key strechting and key strengthening. Password hashing is said to be the most common use of key derivation functions. In key stretching, the key derivation functions are used to stretch the cryptographic keys into longer keys, or to obtain the cryptographic keys of a required format. In key strengthening, the key derivation functions are used to extend the cryptographic key with a random salt (a random number that acts as a cryptographic salt) and then delete the salt in a secure way. Popular key derivation functions are Argon2, Lyra2 and PBKDF.

Argon2

Argon2 refers to a key derivation function that was announced as the winner of PHC (Password Hashing Competition) in 2015. Argon2 is mainly used to hash passwords for key derivation, credential storage, and other applications. Argon2 provides three variants which are Argon2i, Argon2d and Argon2id. Argon2d is designed to resist GPU cracking attacks. Argon2d is designed for password-based key derivation and password-hashing. Finally, Argon2id is a hybrid version (Dinu, 2017.).

Lyra2

Lyra2 refers to a key derivation function and is also called password hashing scheme (PHS). Lyra2 works in a way that it takes a salt and a password as inputs in order to create a pseudorandom output. The pseudorandom output can be utilized as an authentication string or as the key material for cryptographic algorithms (Chen 2009, 2.).

PBKDF

Password-Based Key Derivation Functions (PBKDF1 and PBKDF2) are key derivation functions which are built to reduce the vulnerability of encrypted keys against brute force attacks. PBKDF2 uses a pseudorandom function (e.g. HMAC) and applies it to the input password (or passphrase) along with a salt value. PBKDF2 repeats the same process many times in order to produce a derived key. The derived key generated by PBKDF2 is used as cryptographic key.

In the publication of the IETF in 2017, it has been stated that PBKDF2 is a recommended solution for password hashing (Moriarty, Kaliski & Rusch 2017, 2.).

4.2 Server-side encryption and end-to-end encryption

Server-side encryption is the cryptographic technique that manages your data and the encryption key along with it, encoding the information only when it has been successfully uploaded to the cloud provider. However, the encryption key to encrypt and decrypt is stored together with the data, leaving the data vulnerable for anyone. In some end-user agreement licenses, cloud service providers may agree to keep the data confidential, but can use the data for their own purposes. Therefore, the Cloud Security Alliance advised cloud users to retain complete control over their data (Tietz 2013). The data security and privacy mentioned above can be solved by end-to-end encryption.

Client-side encryption is the cryptographic technique that manages and encrypts the data on the sender’s side. The encryption stage happens before the data is transmitted to a cloud server such as public cloud service provider. The client-side encryption features an encryption key (or a passphrase) which is not available to the cloud provider. Client-side encryption offers a high level of data security and privacy since it allows for the development of zero-knowledge applications (which the providers cannot access).

End-to-end encryption is the cryptographic technique which can be viewed as a specialized use of client-side encryption (Pkware 2018). When data is secured by end-to-end encryption, only the sender and receiver have the right to access to it. End-to-end encryption provides protection for data transmission between two parties (sender and receiver) without the involvement for the third-party. However, generally end-to-end encryption technology encrypted the data during the transmission between users and cloud service providers. In his article, Zafer – CEO at pCloud – stated that client-side end-to-end encryption is the best idea in data security in cloud computing (Zafer, 2016.).

5 CLIENT-SIDE E2E ENCRYPTION IN PUBLIC CLOUD

5.1 Current situation with popular public cloud storage providers

Most of the cloud storage services do not offer client-side encryption. The cloud storage always offers to encrypt the data on the server side. This type of encryption is called server-side encryption, as explained above. Server-side encryption only happens after the cloud storage receives the uploaded data, but before the data is written to disk and stored. The server will provide the default encryption keys, which are the server-side encryption key to encrypt the data; or the users can create and manage their own encryption keys and replace the default ones (Google Cloud 2018a). Client-side encryption, which happens before the data is sent, must be created and managed by users using their own tools (Google Cloud 2018b).

Dropbox shared the comparable situation, even though the file infrastructure is strengthened with multiple layers of protection (includes secure data transfer, network configuration, data encryption…), client-side end-to-end encryption is not provided (Dropbox 2018).

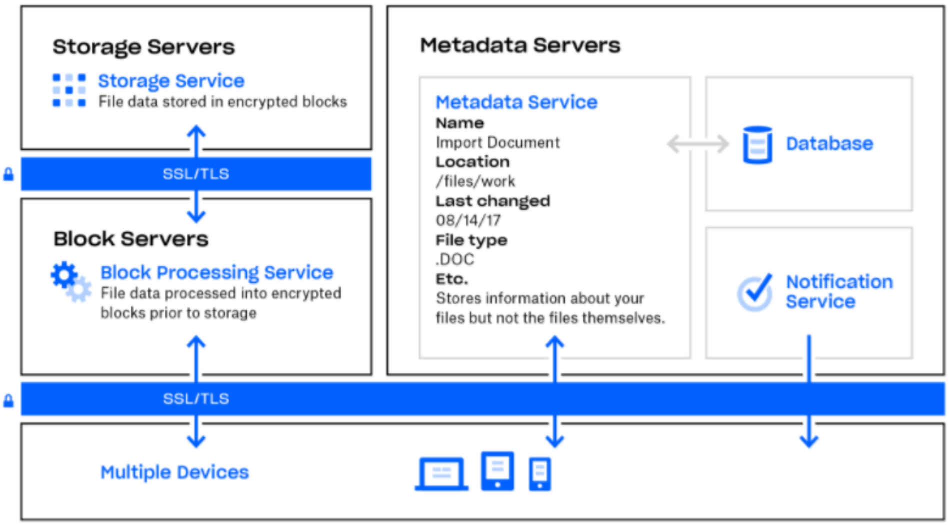


Figure 10. Dropbox's Security Architecture (Dropbox 2018).

On the other hand, iCloud took a step further in building their security technologies, leading the industry by adopting end-to-end encryption. Apple stated that with end-to-end encrypted data, only the users can access to the data through their devices (which has the user's iCloud account signed in); and not even Apple has the right to access to the encrypted information (Apple, 2017.). However, the problem arises is that only Apple devices can be used to access to iCloud. Even though it appears to be an amazing secured solution, the only restriction is the variety of connecting devices.

Other popular cloud storage services with end-to-end encryption are Sync.com and Tresorit. They are both public cloud storage services which offers zero knowledge encryption. Sync.com comes with 5GB free and then 8 dollars per month billed annually. Tresorit does not offer free cloud storage. However, they offer 14-day free trial, then comes 10€ per month billed monthly (or 8,33 euros per month billed annually).

However, when using a public cloud service such as Google Drive, DropBox, Amazon Drive, etc. a third-party tool must be taken in use to encrypt the data on the client-side. Some popular client-side encryption tools are Cloudfogger, Boxcryptor, Cryptomator,etc.

Some popular client-side encryption tools will be taken into testing and analysis below to demonstrate the use of client-side encryption in public cloud server. However, the tests are only for demonstration since the involvement of third-party software defeats the aim of the thesis – which is to provide an integrated solution. Moreover, the test's result will point out that using a client-side encryption tools also has its upsides and downsides.

5.2 **Client-side encryption tools**

Some popular client-side encryption tools are Cloudfogger, Boxcryptor, Cryptomator, SharedSafe, etc. According to the result of the performance analysis of client-side encryption tools carried out in 2014, Cloudfogger was reported to be the best client-side encryption tool at the time (Das, Hossain, Sardar, Biswas & Nath 2014, 897.). However, Cloudfogger project was reported to be shut down in 2016 (Cloudfogger, 2016.).

BoxCryptor's main function is to encrypt the user's sensitive files and folders in more than twenty cloud storages (include Google Drive, Dropbox, OneDrive, etc.) before uploading the files/folders to the cloud storage. BoxCryptor supports multiple platform such as Windows, iOS, Android, … The encryption algorithm used in BoxCryptor comprises of both AES and RSA. Futhermore, BoxCryptor requires authentication and supports offline encryption. The advantages of BoxCryptor are: high security level, cross-platforms, availability for all major cloud storages, encryption and decryption occurs directly on the device (so the password is not transferred anywhere else). However, BoxCryptor limits the devices used to encrypt and decrypt data to only two devices, an upgrade will cost at least 36 euros per year. Furthermore, due to its highly secured encryption key, a folder after being encrypted can approximately double in size (2GB orginally can turn to 4GB in the cloud).

6 CLIENT-SIDE E2E ENCRYPTION IN PRIVATE CLOUD

6.1 Current situation with popular private cloud storage apps

By 2014, ownCloud was the only open-source cloud storage that offered client-side encryption and became popular among enterprises and home users (Salcedo, 2014.). In the recent years, there are many other client-server software established which allows users to create their own file hosting services and use them. These applications offer both client-side end-to-end encryption and server-side encryption, which help to secure data in the cloud. Alternatives to ownCloud, there are many popular cloud storage apps which offer end-to-end encryption such as NextCloud, Seafile and Pydio. In the Google Trends Graph below, it is clear that ownCloud and NextCloud are the most popular private cloud packages in the market.

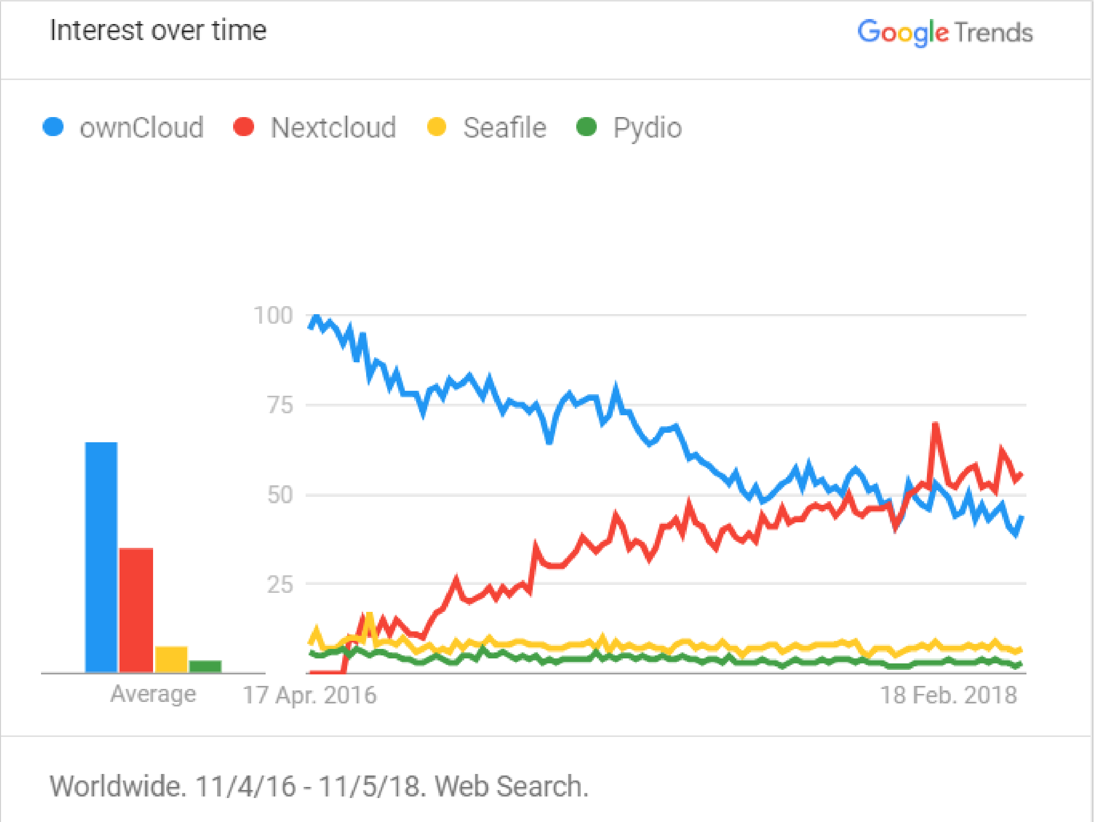


Figure 11. Interest over time in four private cloud storage apps (Google Trends 2018)

Currently, ownCloud offers users the software and support needed to build their own on-site private cloud. ownCloud also gives the users all the controls over their security settings with various built-in modules such as end-to-end encryption, key management, file integrity checking, two-factor authentication, auditability/ logging. OwnCloud has been founded and developed since 2014, therefore, ownCloud is stable and widely adopted by enterprises.

However, compared to NextCloud, there are features which are only available for Enterprise subscriptions only (such as File Access Control) or not offered (Weather, Resource Monitoring, etc.).

Similar to ownCloud, Seafile is a private cloud package which offers file sync and share with high performance and reliability. Seafile also features E2EE encryption and guarantees zero-knowledge encryption. However, Seafile's community edition does not support many features such as file-locking (lock a file to prevent concurrent editing), fine-grained folder permission (set read/write permission on sub-folder), full text search, etc. The pro edition is only offered free up to three users (Seafile, 2018).

Pydio is introduced as an open-source file sync and share software which can be deployed on-premise or in a private cloud. Encrypting data at rest can be done by installing a third-party tool, which is ENCFS (Encrypted Filesystem for FUSE). This software allows users to mount encrypted folders which can be decrypted by a password (which prevents unauthorized access). Pydio comes with two options: the open-source distribution and the enterprise distribution. The open-source distribution is offered with a lot of limitations, such as secured update engine, advanced admin dashboard, etc. (Pydio, 2018)

Compared to the solutions mentioned above, NextCloud is an open-source software which allows users to be productive in file syncing and sharing without losing control. According to figure 11, even though NextCloud has only been founded two years ago, the interest in NextCloud has been growing rapidly and surpassed ownCloud. NextCloud develops its product with full focus on data security, and throughout the data's entire life cycle. In 2018, the ITZBund (German Federal Administration) chose NextCloud to be their secure file exchange solution (NextCloud, 2018b.).

6.2 Introduction to NextCloud

Developed in 2016, NextCloud is a fork of ownCloud project and developed by Karlitschek and other core team members of the ownCloud Inc. NextCloud is a free and open-source suite of client-server software that offers file hosting services. Therefore, anyone is able to install and operate NextCloud on a private server.

In a world with rapidly growing threats around data security and surveillance, the idea of enabling home users and organizations to take control over their own data in the cloud is more important. With NextCloud, the capabilities to host your own cloud, control over your data and any access to your data are granted to users. NextCloud committed to securing the data and they are certain of offering the best security in self-hosted file sync and share. NextCloud follow the best security standard in the industry (ISO27001:2013) and offers some of the highest open-source security bug bounties. NextCloud features a Bug Bounty Program at HackerOne – a vulnerability coordination and bug bounty platform – which is up to five thousand dollars (HackerOne 2017). NextCloud product is designed with compliance in mind. By the end of 2017, NextCloud announced that they were compliant with GDPR and promised to deliver high data security standard (NextCloud, 2017b.).

NextCloud is built around a multi-layered security approach, which comprises of several security layers such as server-side encryption, transport layer security, client-side encryption and two-factor authentication. Furthermore, NextCloud is open-source and can be installed and operated without any costs by home users.

6.2.1 **NextCloud Server-side Encryption**

The NextCloud’s Server-side Encryption is especially designed to support other security layers (such as Transport Layer Security, Client-side End-to-end Encryption and Two-factor Authentication) at the disposal of system administrators.

NextCloud’s Server-side Encryption adopts the most popular, highly proven standards and technologies (e.g. OpenSSL and AES-256) and can be customized to meet the users’ requirements (e.g. custom encryption algorithms, special key management technology, etc.).

MODULAR DESIGN

NextCloud’s Server-side Encryption is designed in a modular way, which provides framework for encryption modules to handle the actual process of encrypting and key handling. Two module options can be chosen, which are the default encryption module (provided by NextCloud) and third-party encryption module. The default encryption module’s functions are to encrypt data at rest, handle encryption keys and enables sharing. The data encrypted by this module are files created and uploaded via WebDAV as well as file versions and files in the trash bin.

THREAT MODEL

According to NextCloud, the server-side encryption was developed specifically to protect from a defined Threat Model, which helps users to decide whether or not it is useful in their specific cases. Firstly, server-side encryption module can be enabled or disable separated for each cloud storage. The name of folders and files are not encrypted; however, these are not leaked due to the use of object storage. The content of files is protected by server-side encryption on the server while at rest as well as on external storage locations. The encryption keys are encrypted with a server key (stored on the server) or a per-user key (not stored on the server). In both cases, NextCloud states that the server gains no access to the data.

ENCRYPTION PROCESS

The process of encryption on the server side consists of generating keys, sharing, encrypting and decrypting files.

GENERATING KEYS

In case of using the server key, only one public / private keypair is generated and utilized to encrypt / decrypt data. On the other hand, there are at least two public / private key pairs are generated (one for every user upon their first login, one for public link shares and potentially one more for recovery) in case of per-user keys.

The public / private key pair is generated by NextCloud when the user logins for the first time. The key pair established is an RSA-4096 key. Moreover, a private key password is generated to encrypt the user's private key. This specific password is made by running the user's login password through a password-based key derivation (PBKDF2 key derivation) with 100,000 iterations.

FILE KEY HANDLING

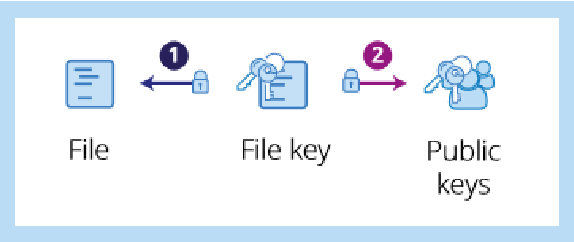


Figure 12. File key handling in Server-side Encryption (Rode 2017, 4).

Each file in NextCloud server is encrypted with its own, unique 256-bit random file key. This specific file key is encrypted against the public keys of the users who have the right to access to the file. The process of encrypting the file key to the users' public keys is done with a share key, which is used to control file access. The share key is 128-bit long and unique for each user.

FILE AND SHARING OPERATIONS

A series of encryption operations occur when accessing or sharing files. These operations are: (1) the file is created and uploaded, (2) access is granted to a user, (3) authorization to access the file is revoked, and (4) authorized user accesses the file. In case of using the server key, the encryption happens only once against the server key. In case of using the per-user keys, the encryption happens against the user keys.

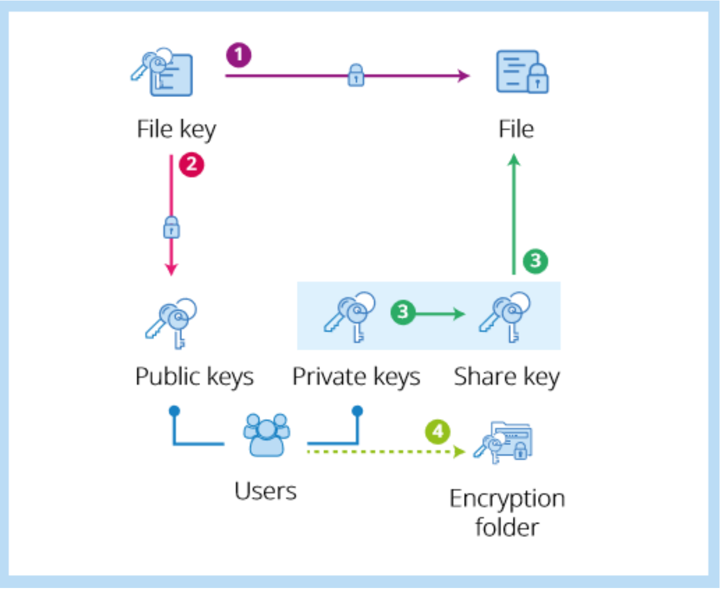


Figure 13. Encryption process when a new file is created and uploaded (Rode 2017, 5).

Firstly, when the file is created and uploaded by a user, it is encrypted by an associated file key, which has been explained above. The file key is encrypted against the authorized users' public keys with the 128-bit share key. The share key can be decrypted using the users' private keys. The file key is stored in the encryption folder.

Secondly, when access is granted to a user, NextCloud will generate a new 128-bit share key by encrypting the file key against all authorized users' public keys.

Thirdly, when the access right is removed from a user, NextCloud will automatically remove the current share key and generate a new one. The new share key will be generated by re-encrypting the file key with the current authorized users' public keys.

Lastly, when an authorized user accesses the specific file, user's private key will be required. The private key has been decrypted when the user logs into the server using login password. The decrypted private key is stored in users' PHP session (Rode 2017, 4.).

6.2.2 NextCloud End-to-end Encryption

REQUIREMENTS

NextCloud End-to-end Encryption is designed to fulfil specific business and technical criteria, which are security properties, widely-used and successfully-tested libraries for cryptographic primitives, sharing functionality, central data recovery, multi-device management, authenticated key exchange, and versioning.

The end-to-end encrypted folders must use an encryption scheme that guarantees the confidentiality, integrity and authenticity of the data. Only the authorized users will have access to the folders. One must not be able to interfere with the data even with the writable access to the ciphertext to maintain the integrity of data. For instance, when the specific encrypted folder has been removed from the file system, but can still be found in the data, warning should be given to the user. Authenticity of the encrypted files must be guaranteed. The access to the encrypted data (ciphertext) must not leak any information on the files names, content or the directory structure. Moreover, the users' public keys must be auditable.

The libraries used for cryptographic primitives must have been through successful security audits and been available and used widely. The libraries used must also be available for a variety of environments, such as: Windows 7+, PHP 7.0+, Linux distributions, Mac OS X 10.9+, Android 6.0+ and iOS 9+.

NextCloud End-to-end Encryption must allow sharing encrypted folders with other users. However, sharing single files from an encrypted folder or sharing the encrypted folders with the groups are considered out-of-scope. However, this feature is a work-in-progress and has not yet been implemented in the latest testing release.

End-to-end encryption works in the way that encryption keys are generated to encrypt the data during transfer. These encryption keys help to prevent un-authorized access or even access from the server. However, the downside of end-to-end encryption is that users may forget or lose their encryption keys, results in data loss. Therefore, an optional central data recovery is offered with the following functions. The central recovery key per instance can be created and safely exported to be stored elsewhere from the instance itself (e.g. on a physical vault). All the user's data will be encrypted to the enabled central recovery key.

Users must be able to access the encrypted data easily from any device (desktop devices and mobile devices). NextCloud designed the key exchange system to follow the properties which are: key exchange process between different parties must be frictionless; and exchange keys must also be auditable. Additionally, in case of future changes in cryptographic handling or metadata, the protocol must support versioning (NextCloud, 2017a).

TECHNICAL IMPLEMENTATION

NextCloud End-to-end Encryption is implemented based on an asymmetric cryptographic system. Every user will have exactly one public/private keypair.

The technical implementation consists of five parts, which are (1) create and sync identity, (2) encrypt folders and add files to encrypted folders, (3) access encrypted files, (4) share encrypted folders, and (5) remove users from accessing the encrypted folders. As I have mentioned above, the sharing and removing users from accessing the encrypted folders are works in progress and are not available in the latest testing release.

CREATE AND SYNC IDENTITY

The create and sync identity process consists of initial device, sync identity, and add further devices.

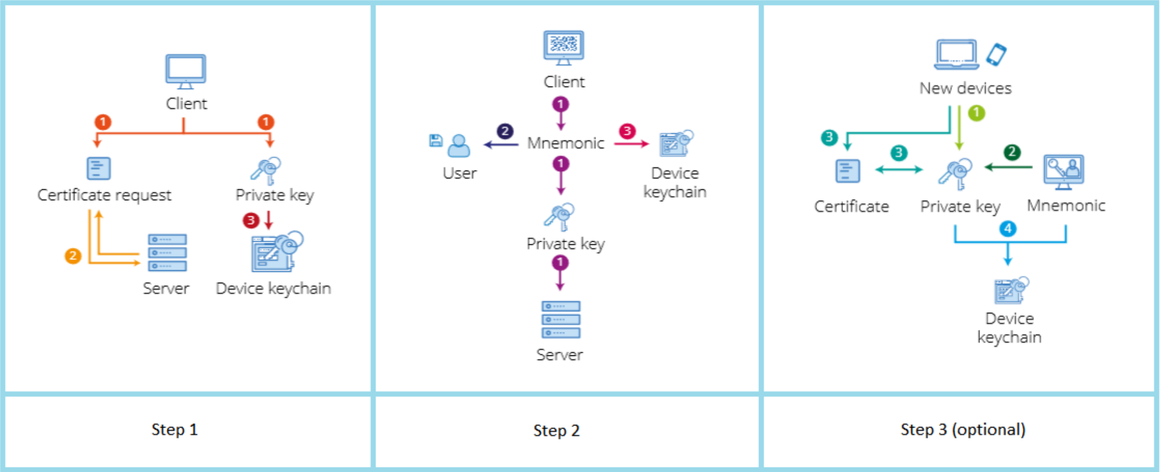


Figure 14. Create, sync and add device (NextCloud 2017a)

The first step the client needs to do is generating a certificate request (X.509 certificate) and a private key. Secondly, the certificate request will get signed by the server if the CN of user matches the current user ID. After the client receives the signed certificate by the server, the private key and public key will be stored in the keychain of the device (NextCloud, 2017a).

After that, the client will encrypt the private key generated in the first step with a 12-word long mnemonic (from the English BIP-0039-word list). The BIP-0039-word list contains 2048 different words, which makes 2048^12 possible key combinations for the mnemonic. The private key is encrypted in the client side by using AES/GCM/NoPadding as cipher (128-bit key), PBKDF2WithHmacSHA1 (PBKDF2 with HMAC SHA-1) as key derivation, and the generated mnemonic as password. The encrypted X.509 private key will be uploaded to the server in order to simplify the addition of more devices. The mnemonic key will be displayed to users and users are asked to store it. The mnemonic is also stored in the keychain of the device (NextCloud, 2017a).

In case of adding further devices, meaning that the signed certificate has already been available for user, the client only has to download the private key from the server. The user will be asked for the mnemonic and decrypts the private key (using AES/GCM/NoPadding and PBKDF2WithHmacSHA1). If the private key belongs to the previously downloaded certificate, the private key will be store in the keychain of the device (NextCloud, 2017a).

ENCRYPT FOLDERS AND ADD FILES

The following process consists of three steps: (1) create an E2E encrypted folder, (2) upload a file into the E2E encrypted folder, and (3) update the existing files. Data access to this kind of folder happens via WebDAV API.

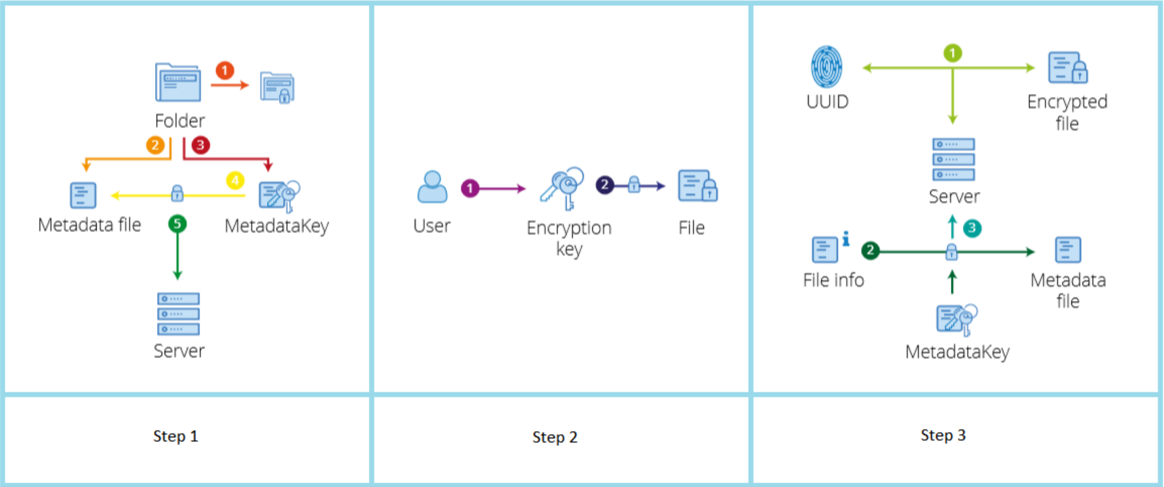


Figure 15. Encrypt folders and add files (NextCloud 2017a).

The first step of the process is to create a folder via WebDAV and mark the folder as end-to-end encrypted. Only empty folders can be flagged as E2E encrypted. Once the folder has been marked, the folder is no longer accessible via the web. Each folder to be marked E2E encrypted contains a metadata file (consists if metadata of files, access list to the folder, and key materials for files in the folder) and metadata key. The metadata key elements are encrypted to all the public keys that have access to the E2E encrypted folder. The metadata key is used to encrypt all of the values in the metadata file. And the encrypted metadata file is stored on the server (NextCloud, 2017a).

The second step is to add files to the E2E encrypted folder. In case a file is uploaded to the encrypted folder, the client will generate a new 128-bit encryption key (using AES/GCM/NoPadding) for the file. It also creates a random identifier (e.g. UUID) for the file and upload load the encrypted file with the random identifier as the file ID. The encrypted file will be upload via WebDAV. The file will be added to the files array in the metadata file (NextCloud, 2017a).

If an existing file in the encrypted folder is updated, the client creates a new 128-bit encryption key and encrypt the file using AES/GCM/NoPadding. After that, the client will lock down the encrypted folder and use the existing identifier to upload the file via WebDAV. The file will be updated in the files array in the metadata file and encrypted with metadataKey. The metadata file is also updated and encrypted with metadataKey (NextCloud, 2017a).

ACCESSING ENCRYPTED FILES

The figure below explains the steps the client has to take when accessing an encrypted file.

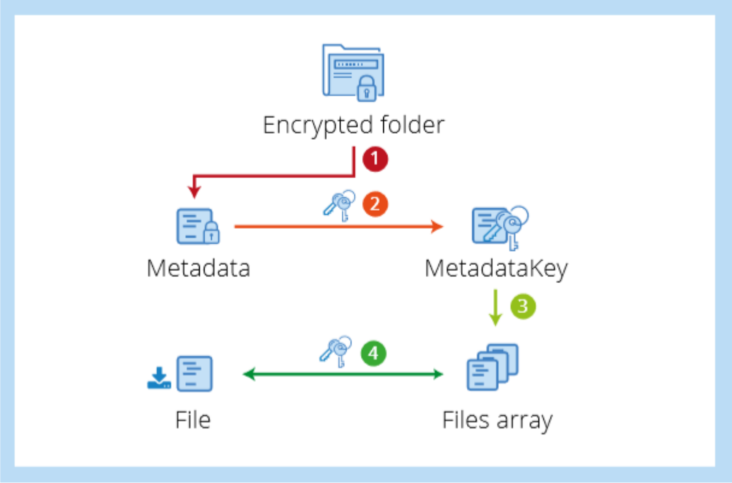


Figure 16. Accessing encrypted files (NextCloud, 2017a.)

Firstly, the client will download the metadata of the encrypted folder. The user’s private key is used for decrypting the metadataKey. Then, the client will loop through the files array in metadata file and decrypt the array with the latest metadataKey. Lastly, the files will be downloaded via WebDAV and decrypted using AES/GCM/NoPadding and the referenced file keys in the file array (NextCloud, 2017a.).

SHARING ENCRYPTED FOLDERS

In order to share E2E encrypted folders, the clients will go through these steps below to build a trust relationship between users. However, this feature is a work-in-progress and has not yet been implemented in the latest testing release.

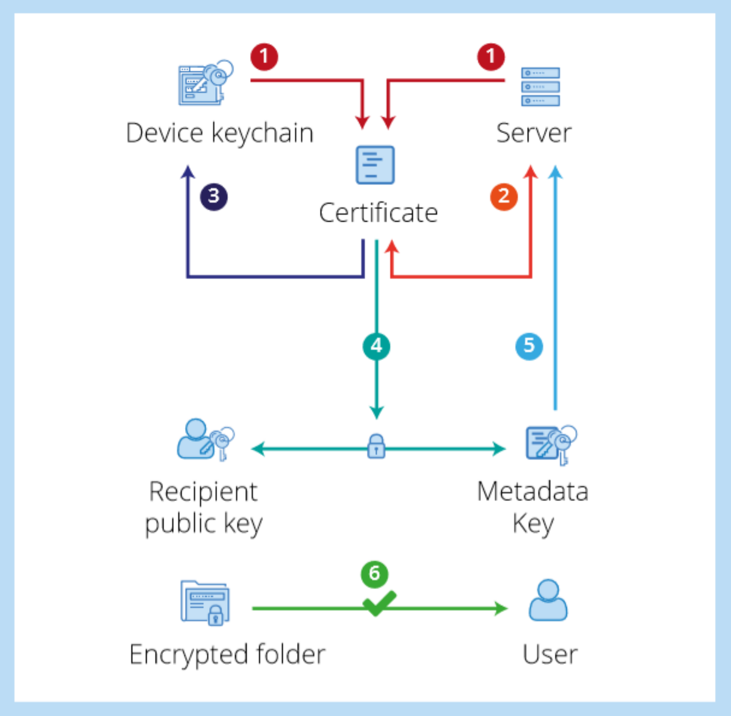


Figure 17. Sharing encrypted folders (NextCloud, 2017a.).

Firstly, the client will check if there is a certificate for the userID has been already downloaded. If there is, then the available certificate will be use. If there is none to be found, then the client will query for the user certificates. After the client receives the certificate, it verifies if the certificate has been signed by the server public key. If the certificate is verified, it will be use to establish a trust relationship. If the certificate is not issued by the server public key, a warning will be shown stating that the share is not possible to the user. The user certificate will be store in the keychain of the device for the next operation.

After the certificate is successfully verified, the metadataKey will be re-encrypt to the new recipient public key and the new recipient is added to the sharing array. The metadata file is updated and uploaded to the server. The E2E encrypted folder will now be shared with the new recipient through OCS share API (NextCloud 2017a).

REMOVE USER FROM ACCESSING ENCRYPTED FOLDER

In order to remove a recipient from the existing share list, the following steps need to be taken. However, this feature is a work-in-progress and has not yet been implemented in the latest testing release.

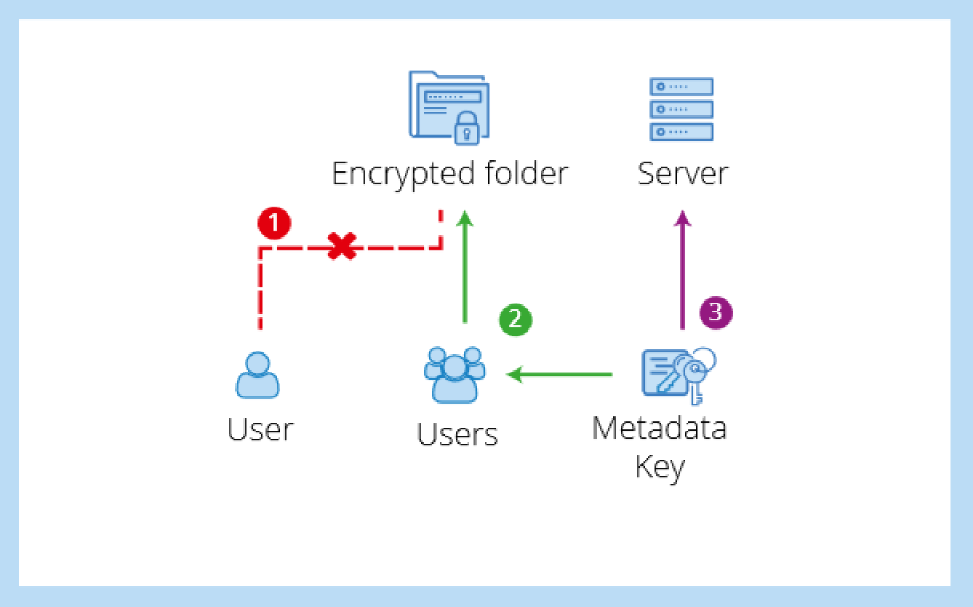


Figure 18. Remove user from existing share (NextCloud 2017a).

The client will first unshared the folder via the OCS share API to the recipient. Then, a new metadataKey must be re-generated. The removed recipient is now deleted from the sharing array. The new metadataKey is re-encrypted to the public keys of current recipients. The metadata file is updated and uploaded to the server (NextCloud 2017a).

7 IMPLEMENTATION PLAN

7.1 Chosen implementation method

The thesis was designed as a mix of comparative and experimental research in order to proof the thesis 's concept. The audience scope of the research is mainly home users.

The implementation method chosen is testing end-to-end encryption on a private cloud server. The private cloud server will be built with NextCloud. The experiment is expected to give positive results in securing data in a private cloud server with both client-side end-to-end encryption and server-side encryption.

7.2 Rationale for implementation method

Performing an experiment on private cloud server is required to provide readers a prove of concept that data security in private cloud is on a prominent level. The experiment will not only be the prove of concept but also a guideline for readers who desire to build and control their own private cloud server.

7.3 Application of method

The cloud solution to be built is an on-site private cloud server. Comparing to the available SaaS, PaaS or IaaS solutions on the market, the experiment not only shows a way to build a private cloud server solution, but also gives an insight into all layers needed in any cloud service. Users are fully responsible for all layers, such as physical environment, network, infrastructure, operating system, user access, data and application. All of the layers will be covered through the project phases as follow.

7.3.1 Project phases

Since the thesis is a proof of concept research. An experiment on end-to-end encryption in private cloud server will be carried out to prove the research's hypothesis. The experiment will be divided in three phases which are: installation, testing and monitoring.

The installation stage consists of installing Apache2, PHP and the installation of NextCloud in Apache2. The first phase also includes other set-ups, which are: moving data folder on the server side, increase max upload size, allowing .htaccess override, SSL set-up, and upgrading NextCloud.

The second phase of the experiment is testing. The server side will be tested first to check if the private cloud server is up and running. Performing tasks such as upload, download, enable encryption modules will be carried out. After the server side is stably up and running, the desktop client-side will be downloaded and set-up. Syncing folders and end-to-end encryption will be primarily tested. On the server side, access to the folder which is encrypted on the client side will be experimented to check if the server has the access right.

The last phase of the experiment is monitoring. After testing the private cloud server, network monitoring will be performed to check the level of data security and privacy. The result of the experiment is expected to give positive results in the level of data security and privacy in private cloud server. The result of experiment can also be used as the guideline for building your own private cloud server at home.

A user test will be performed and reported in the end of the implementation in order to test and evaluate the performance and the built cloud solution.

7.3.2 Working environment

Working environment:

- Hardware needed:
  - Raspberry Pi 3
  - SanDisk MicroSD Card 16GB
  - Router: 300M Wireless N Router, Model No. TL-WR841N / TL-WR841ND
  - Laptop: Acer Aspire F5-573G
    - Processor: Inter® Core™ i5-7200U CPU @ 2.50GHz 2.70 GHz
    - RAM: 8.00 GB
    - System: 64-bit operating system, x64-based processor
- Operating system: Raspbian Stretch
- Software: NextCloud, WireShark

Network setup in local network

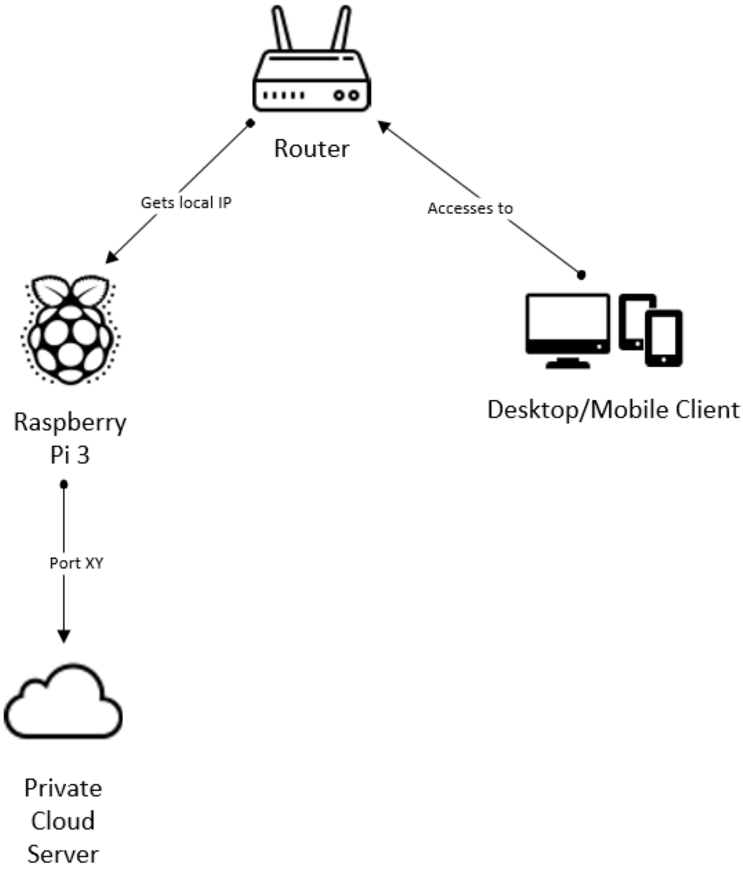


Figure 19. Private Cloud Server Network Setup (inside the local-area network)

In case of inside the local network, the client will access to the private cloud server by connecting to the Raspberry Pi 3 under the same local-area network (LAN). The desktop client will access to the router and the router's responsibility is getting the local IP of the Raspberry Pi.

Network setup outside local network

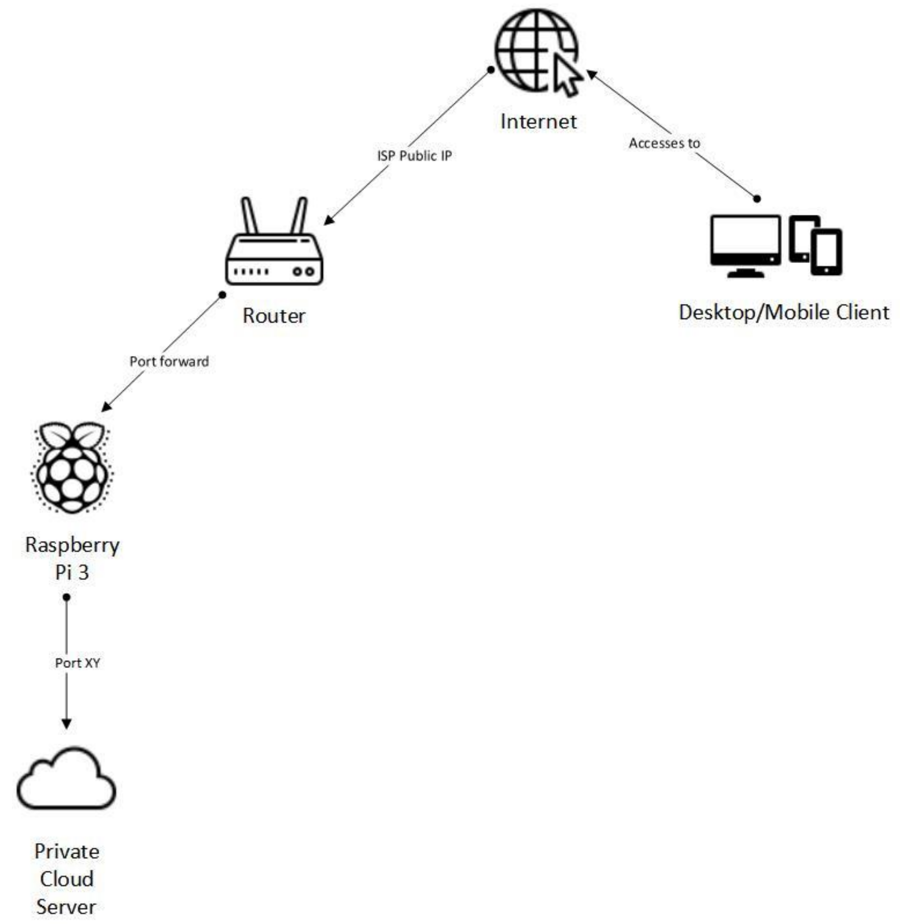


Figure 20. Private Cloud Server Network Setup (outside local-area network)

After the cloud server has been port-forwarded, the cloud server can be Accessed from the internet. The client can access to the server at anytime and from anywhere with the internet connection. The process of port-forwarding the cloud server will be explain in the empirical part below.

8 BUILDING A PRIVATE CLOUD SERVER WITH NEXTCLOUD

8.1 Installation

The private cloud server established will run on a Raspberry Pi 3, which runs Raspbian Stretch operating system. The server will be port-forwarded to the router and enable clients to access through the Internet by calling the ISP Public IP of the router.

The first step which needs to be done is connecting to the Raspberry Pi. In order to connect to the Raspberry Pi with a command line client, SSH needs to be enabled (Appendix 1). SSH (Secure Shell) refers to the cryptographic network protocol which can be used to operate network services securely.

Starting SSH server on port 22 will allow running command line on the Raspberry Pi. The first step is accessing to the Raspberry Pi software configuration tool (raspi-config) by running the following command.

```
# sudo raspi-config
```

After that, SSH can be enabled in the Raspberry Pi software configuration tool as described in Appendix 1, Image 1 and Image 2. After SSH server is enabled on the Raspberry Pi, any SSH client can connect to the Raspberry Pi (under the same local-area network). The established of the connection to Raspberry Pi 3 is done with a free SSH client (PuTTY) (Appendix 1, Image 4). After login is succeeded, it is possible to run command line on the Raspberry Pi (Appendix 1, Image 5).

8.1.1 Pre-installation

Step 1: Update the package repositories

Step 2: Install Apache HTTP Server – the most popular, free and open source cross-platform web server (Appendix 2, Image 6).

Step 3: Install PHP7.0 and its packages. PHP (or Pre-processor) is a server-side scripting language, which is not only especially designed for web developing but also a general-purpose programming language (Appendix 2, Image 8).

8.1.2 NextCloud Installation

Step 1: Install NextCloud in Apache2 HTTP Server (Appendix 3, Image 10).

Step 2: Setting directory and access rights. This step consists of creating a new data directory for NextCloud to operate in, granting the control over data folder for the correct user and group, giving right permissions and giving the data group control over the configuration and application folder (Appendix 3, Image 11).

After the two steps, the installation process of NextCloud can be started by going to the Raspberry Pi's IP address plus "/nextcloud". In this case, the IP address of the Raspberry Pi is 192.168.43.30.

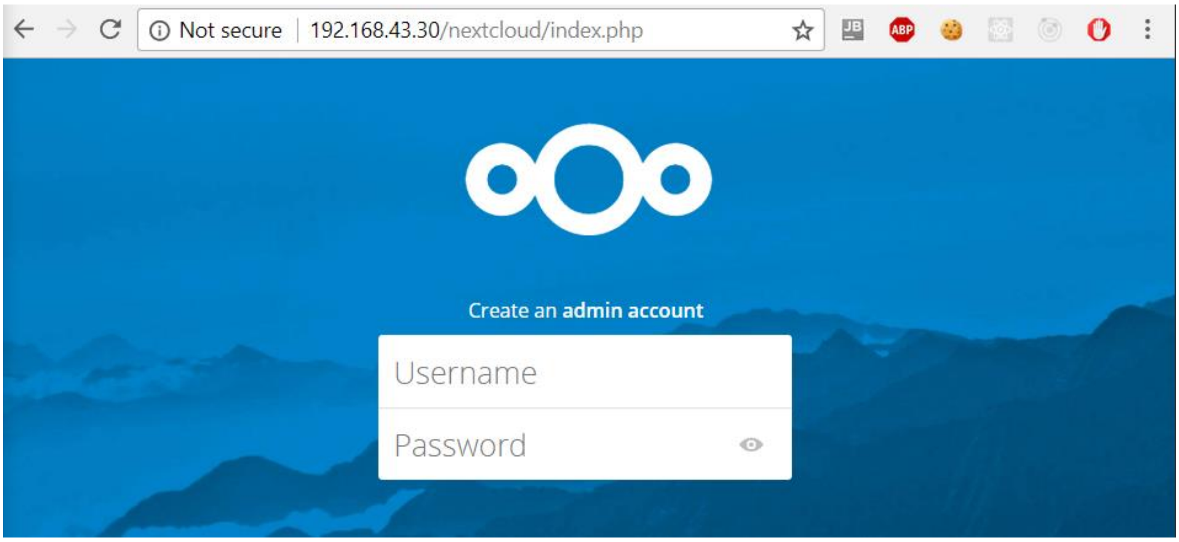
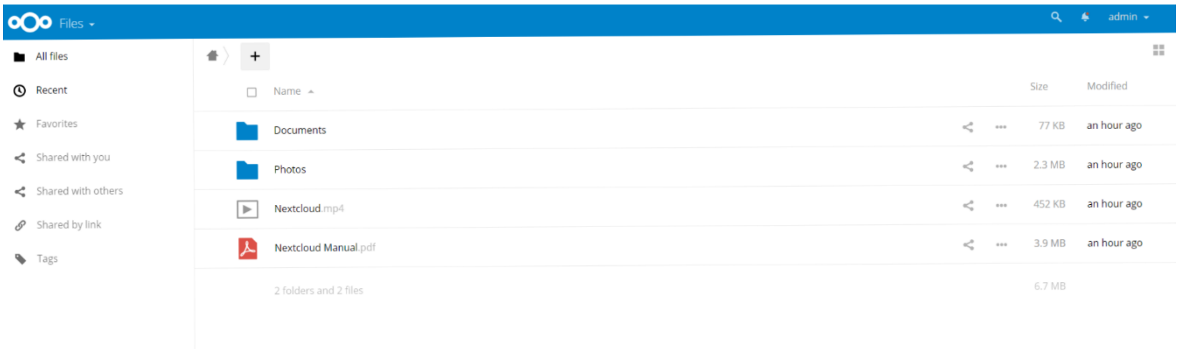


Figure 21. NextCloud up and running.

In order to complete the installation, username and password need to be entered to create an admin account. After creating the admin account, one can see the Raspberry Pi NextCloud interface.



Name	Size	Modified
Documents	77 KB	an hour ago
Photos	2.3 MB	an hour ago
Nextcloud.mp4	452 KB	an hour ago
Nextcloud Manual.pdf	3.9 MB	an hour ago
2 folders and 2 files	6.7 MB	

Figure 22. NextCloud's interface.

8.1.3 **Other Setups**

Step 1. Moving the data directory on the server-side.

When the data directory is created, it was placed in the web accessible directory. Therefore, moving the data directory on the server-side is needed to have a better secured and usable data directory. The same process can be applied when moving the data directory from the Micro SD Card to a larger external hard-drive when more storage space is needed. The process of moving the data directory is explained in Appendix 4, Step 1, Image 13 and 14.

Step 2. Increase the max upload size.

The PHP installed provided a low upload limit by default (2MB). To change the default upload limit, php.ini file needs to be modified. Hereby, the upload max file size is set to 1024M, but it can be changed to the maximum file size one will upload to NextCloud. The process of increasing the max upload size is mentioned in Appendix 4, Step 2, Image 16 and 17.

Step 3. Allowing the .htaccess override

The distributed configuration files (.htaccess files) allows making config changes on a per-directory basis. The .htaccess files should only be utilized when the content-providers need to change the configurations to the server on a per-directory basis. The .htaccess override can be enabled in apache2 config file (Appendix 4, Image 18).

Step 4. SSL Setup

Secure Sockets Layer (SSL) is the standard security technology for building an encrypted link between the server and the browser. The SSL connection can be established when the SSL Certificate is created. Typically, when one applies for the SSL Certificate, the SSL Certificate will consist of their own domain main, company name, address, city and country. In this case, the thesis write does not have a domain name. Therefore, a self-signed certificate will be created. The process of creating a self-signed certificate, enabling SSL and enforcing SSL in order to redirect HTTP traffic to HTTPs traffic are explained in Appendix 4, Step 4.

Step 5. Upgrading NextCloud

Since NextCloud version 11.0.2 was released in the beginning of 2017, there are a lot of missing functions compared to the latest update. Therefore, an upgrade is needed. NextCloud can be upgraded in the Updater from Admin Settings (Appendix 5, Image 26).

8.1.4 **NextCloud Client Installation**

For the desktop client, NextCloud desktop client version 2.5.0.665 will be utilized. This is a pre-release version dedicated for tech preview only. After downloading the pre-release version from NextCloud's main website, the cloud server can be Accessed through the server address (Appendix 6, Image 31).

After entering the domain address and processing through user credentials (Appendix 6, Image 33), users can sync their files from the server with syncing options (e.g. sync everything, sync manually, file destination, etc.) (Appendix 6, Image 34).

8.1.5 **Port-forwarding**

The NextCloud server can be port-forwarded through the router (Appendix 7, Image 35). For security reason, the router's public IP address will be hidden in this thesis. When entering the NextCloud server through the router's public IP address, NextCloud prevents the access through the untrusted domain (Appendix 7, Image 36). The domains can be added as trusted domains manually in the config/config.php file as in Appendix 7, Image 37. After being added as the trusted domain, external access is granted through the public IP address (Appendix 7, Image 38).

8.2 **User test**

8.2.1 **Testing server-side and client-side**

Only the cloud server, only the admin can enable and disable the modules in the cloud server's applications. Therefore, authentication will be required in order to activate server side and end to end encryption (Appendix 8, Image 40).

After enabling Default Encryption Module and End-to-end Encryption, a testing folder will be created on the server-side and tested in both sides (Appendix 8, Image 42). The test folder must be empty to be marked as end-to-end encrypted.

After the testing folder is created on the server-side, it can be end-to-end encrypted on the client-side as explained in Appendix 8, Image 43. After the folder being encrypted on the client-side, the mnemonic will be handed to the user, the user is asked to note the mnemonic in order to use it to decrypt data later on. Therefore, the mnemonic needs to be handled with care since it will be needed when adding a further device or sharing encrypted folders with other users. (Appendix 8, Image 44).

The roles of the mnemonic passphrase are not only to decrypt the private key, but also to create zero-knowledge encryption, which means that the cloud server does not have the capability to know what the data is. End-to-end is enabled on the client-side so that the data can be synced seamlessly and securely between client's devices, without being decrypted in the server-side. After the data is encrypted on the desktop client with the encryption passphrase provided, the server can no longer access to the encrypted file (Appendix 8, Image 46).

The result of this security test is to prove that even if the server got hacked by outsiders, the data on the client-side will be secured.

8.2.2 Testing upload and download speed

The private cloud solution is expected to offer fast performance. Therefore, tests have been performed on uploading and downloading testing files (3.91MB and 1GB). The tests were done inside the local network and outside the local network. The local network speed test is the home connection speed. The mobile hotspot connection is used in demonstrating outside the local network.

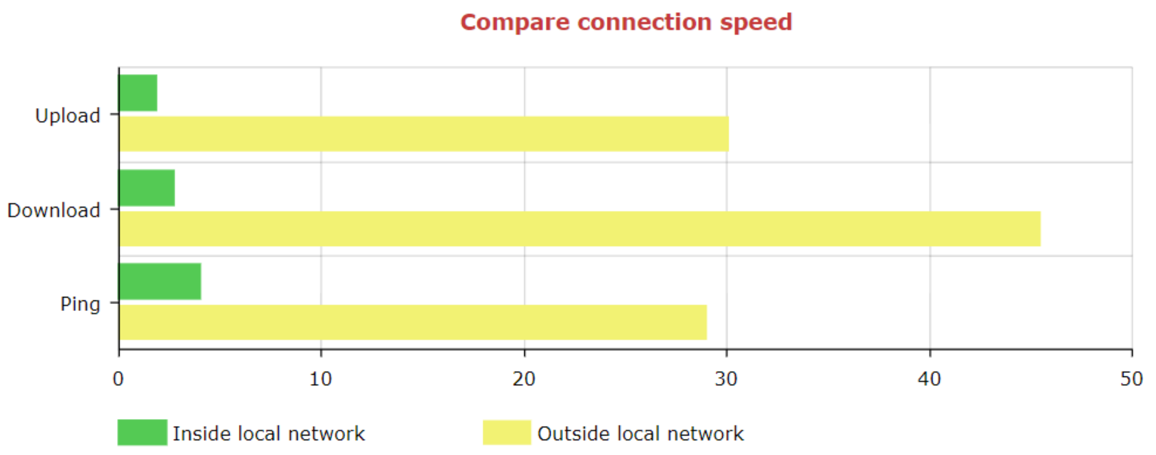
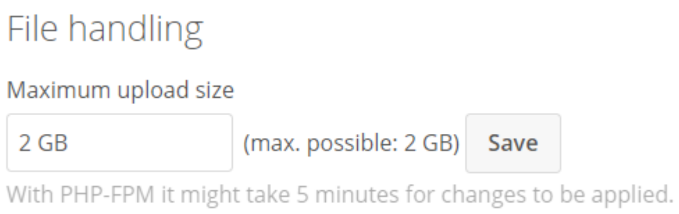


Figure 23. Compare connection speed of inside and outside the local network.

According to figure 23, it can be seen that the local network's speed is much slower than the mobile hotspot's speed.

In order to upload bigger file, maximum upload size has to be changed, this can be done in the web server or using command line to change the php config file.



File handling

Maximum upload size

2 GB (max. possible: 2 GB)

With PHP-FPM it might take 5 minutes for changes to be applied.

Figure 24. Change upload max size on the web server

The chart below gives the benchmarking results in downloading and uploading a PDF file (3.91MB). All of the screenshots of the result can be found in Appendix 8.

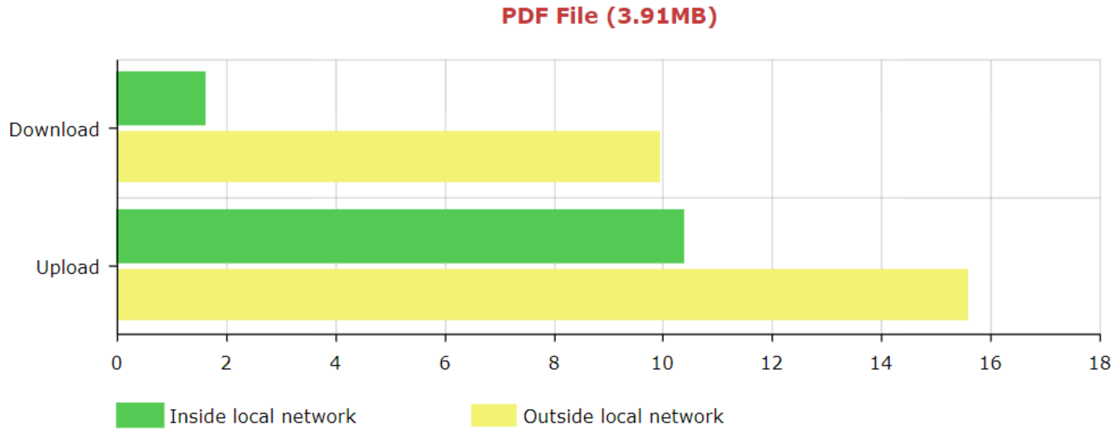


Figure 25. Downloading and uploading a PDF file inside and outside the local network. (Time in Seconds)

The chart below gives the benchmarking results in downloading and uploading a 1GB file from inside and outside the local network.

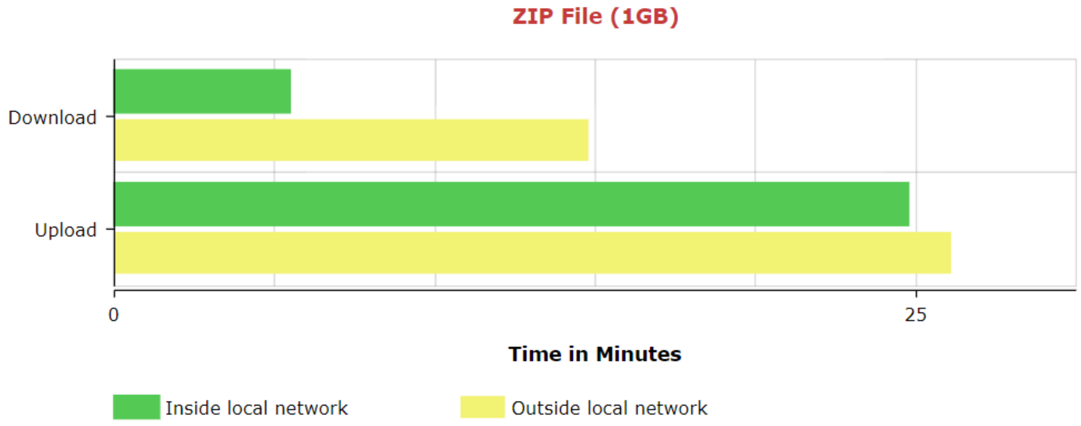


Figure 26. Downloading and uploading a 1GB Zip file inside and outside the local network (Time in Minutes).

The results in figure 25 and figure 26 show that uploading and downloading inside the local network is faster than outside the local network, even though the local network's speed is much slower than the mobile hotspot's speed. Therefore, the performance of the private cloud solution is better within the local network. All of the images and screenshots related to this test are in Appendix 8, Image 47 to Image 56.

8.2.3 **Testing file sharing and dropping**

File sharing and dropping can be done easily and securely in NextCloud server. In order to share and control the access to the shared file, File Access Control module can be enable in the Apps (Appendix 8, Image 57). The File Access Control module helps the admin to be sure that all interactions within the server follow the rules and requirements regarding passwords as well as expiration dates.

The file can be shared directly to users or by a share link, and the file can be set with a password in order to protect its confidentiality (Appendix 8, Image 58). Moreover, an expiration date can be set so that the file will not continue to be shared after the set date. After setting the password and expiration date (Appendix 8, Image 59), the shared file can be Accessed only with the protection password (Appendix 8, Image 60).

Moreover, a secure upload point can be created in the same manner. In order to create an upload link, the admin can select any folder as the destination for the customer to upload by enabling Secure drop (Upload Only). This action not only makes the chosen folder as the file upload destination for the customer, but also hides the existing content of the folder from the shared customer. After the secure drop is enabled, the customer/client can upload files to the server in a secure way (Appendix 8, Image 61).

Additionally, different permissions can be granted to the shared users and different shared users can receive their unique sharing links and their own passwords and expiration date (Appendix 8, Image 62).

8.2.4 **Testing two-factor authentication**

Another security module provided by NextCloud is Two-factor Authentication. The module can be enabled in the Apps under the name TOTP TwoFactor. After that, TOTP application should be installed om mobile device. The TOTP application installed on the user's smartphones or mobile devices in order to generate a one-time password to be checked by the server.

The two-factor authentication settings can be found in Personal Settings > Security > Enable TOTP Authentication. In order to enable two-factor authentication, a backup password needs to be set in case the device is broken or stolen (Appendix 8, Image 63). After that, the QR Code will be showed. In order to activate the TOTP application on the smart device, the QR Code provided by the server should be scanned (Appendix 8, Image 64).

The QR code can be scanned with the TOTP mobile application and a given token number will be showed on the mobile application. Enter the given number into NextCloud server to complete the activating process. However, screenshot showing the token number is not allowed to be taken on the TOTP mobile application.

After two-factor authentication module has been successfully enabled, the user must enter the generated number from the TOTP mobile application (newly generated every 20 seconds) to access the cloud server (Appendix 8, Image 65).

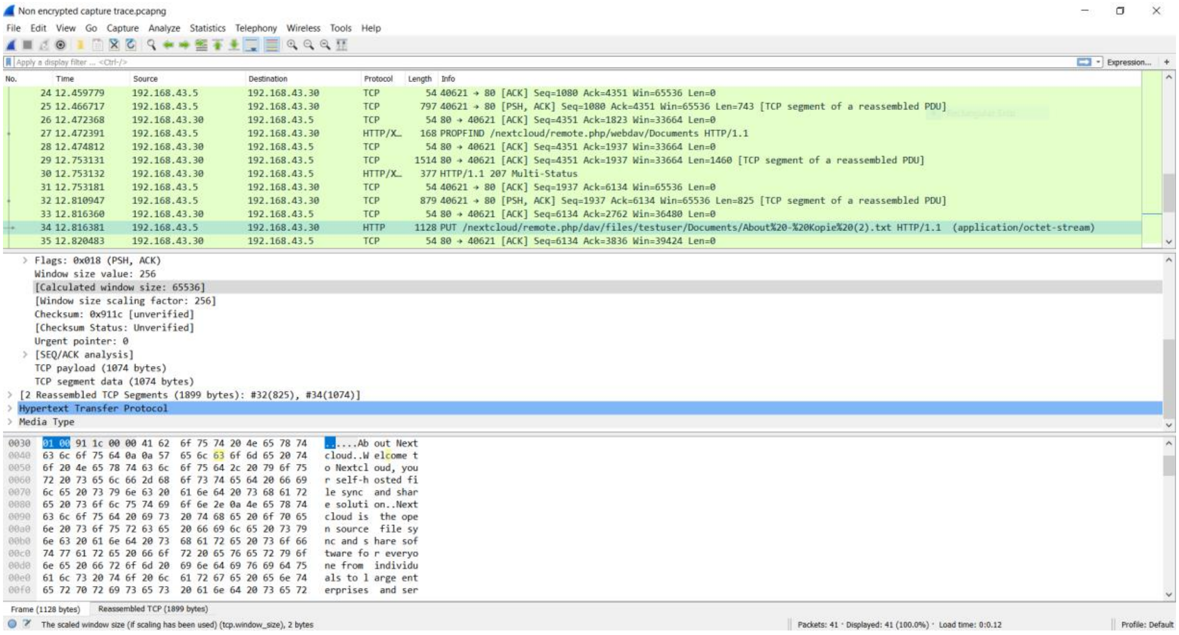
8.3 Network traffic monitoring

In order to perform network traffic monitoring, SSL has to be disabled from the server side. Since SSL creates a secured channel for the network, network traffic could not be tracked if SSL is working. SSL can be disabled, and Apache can be restarted using the follow command lines:

```
# sudo a2dismod ssl
# sudo service apache2 restart
```

To monitor the network traffic, Wireshark – the world’s most widely-used network protocol analyzer – is taken in use.

The network monitoring was performed with a testing text file in the Test E2EE secured folder. The text file has the content of the NextCloud’s general information. The figure below is the network traffic monitoring when the file is not end-to-end encrypted. During the time of transferring from the client to the server, the testing text file is showed in its original form.


Non encrypted capture trace.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...

```
No.  Time       Source          Destination     Protocol Length Info
  24 12.459779  192.168.43.5    192.168.43.30   TCP        54 40621 → 80 [ACK] Seq=1080 Ack=4351 Win=65536 Len=0
  25 12.466717  192.168.43.5    192.168.43.30   TCP       797 40621 → 80 [PSH, ACK] Seq=1080 Ack=4351 Win=65536 Len=743 [TCP segment of a reassembled PDU]
  26 12.472368  192.168.43.30   192.168.43.5    TCP        54 80 → 40621 [ACK] Seq=4351 Ack=1823 Win=33664 Len=0
  27 12.472391  192.168.43.5    192.168.43.30   HTTP/X…   168 PROPFIND /nextcloud/remote.php/webdav/Documents HTTP/1.1
  28 12.474812  192.168.43.30   192.168.43.5    TCP        54 80 → 40621 [ACK] Seq=4351 Ack=1937 Win=33664 Len=0
  29 12.753131  192.168.43.30   192.168.43.5    TCP      1514 80 → 40621 [ACK] Seq=4351 Ack=1937 Win=33664 Len=1460 [TCP segment of a reassembled PDU]
  30 12.753132  192.168.43.30   192.168.43.5    HTTP/X…   377 HTTP/1.1 207 Multi-Status
  31 12.753181  192.168.43.5    192.168.43.30   TCP        54 40621 → 80 [ACK] Seq=1937 Ack=6134 Win=65536 Len=0
  32 12.810947  192.168.43.5    192.168.43.30   TCP       879 40621 → 80 [PSH, ACK] Seq=1937 Ack=6134 Win=65536 Len=825 [TCP segment of a reassembled PDU]
  33 12.816360  192.168.43.30   192.168.43.5    TCP        54 80 → 40621 [ACK] Seq=6134 Ack=2762 Win=36480 Len=0
  34 12.816381  192.168.43.5    192.168.43.30   HTTP     1128 PUT /nextcloud/remote.php/dav/files/testuser/Documents/About%20-%20Kopie%20(2).txt HTTP/1.1  (application/octet-stream)
  35 12.820483  192.168.43.30   192.168.43.5    TCP        54 80 → 40621 [ACK] Seq=6134 Ack=3836 Win=39424 Len=0
```

```
    > Flags: 0x018 (PSH, ACK)
      Window size value: 256
      [Calculated window size: 65536]
      [Window size scaling factor: 256]
      Checksum: 0x911c [unverified]
      [Checksum Status: Unverified]
      Urgent pointer: 0
    > [SEQ/ACK analysis]
      TCP payload (1074 bytes)
      TCP segment data (1074 bytes)
> [2 Reassembled TCP Segments (1899 bytes): #32(825), #34(1074)]
> Hypertext Transfer Protocol
> Media Type
```

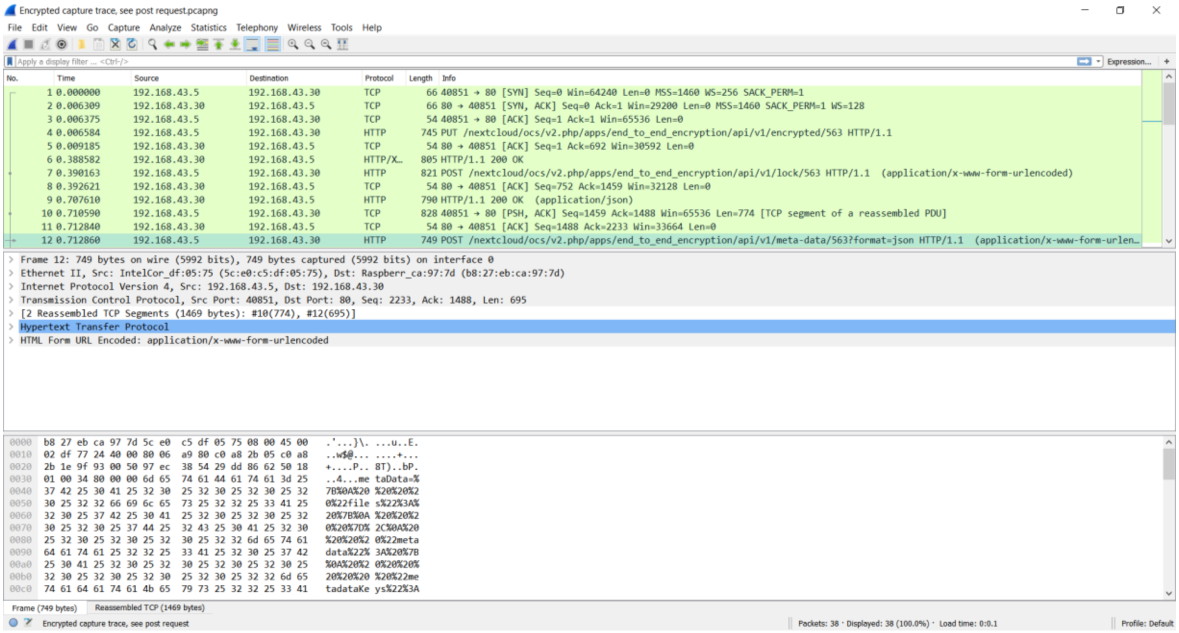
```
0030  01 00 91 1c 00 00 41 62  6f 75 74 20 4e 65 78 74   ......Ab out Next
0040  63 6c 6f 75 64 0a 0a 57  65 6c 63 6f 6d 65 20 74   cloud..W elcome t
0050  6f 20 4e 65 78 74 63 6c  6f 75 64 2c 20 79 6f 75   o Nextcl oud, you
0060  72 20 73 65 6c 66 2d 68  6f 73 74 65 64 20 66 69   r self-h osted fi
0070  6c 65 20 73 79 6e 63 20  61 6e 64 20 73 68 61 72   le sync  and shar
0080  65 20 73 6f 6c 75 74 69  6f 6e 2e 0a 4e 65 78 74   e soluti on..Next
0090  63 6c 6f 75 64 20 69 73  20 74 68 65 20 6f 70 65   cloud is  the ope
00a0  6e 20 73 6f 75 72 63 65  20 66 69 6c 65 20 73 79   n source  file sy
00b0  6e 63 20 61 6e 64 20 73  68 61 72 65 20 73 6f 66   nc and s hare sof
00c0  74 77 61 72 65 20 66 6f  72 20 65 76 65 72 79 6f   tware fo r everyo
00d0  6e 65 20 66 72 6f 6d 20  69 6e 64 69 76 69 64 75   ne from  individu
00e0  61 6c 73 20 74 6f 20 6c  61 72 67 65 20 65 6e 74   als to l arge ent
00f0  65 72 70 72 69 73 65 73  20 61 6e 64 20 73 65 72   erprises  and ser
```

Frame (1128 bytes) | Reassembled TCP (1899 bytes)

The scaled window size (if scaling has been used) (tcp.window_size), 2 bytes | Packets: 41 · Displayed: 41 (100.0%) · Load time: 0:0.12 | Profile: Default

Figure 27. Non-end-to-end encrypted capture trace.

After enabling end-to-end encryption in the client-side, network traffic tracking and monitoring is carried out. The figure below shows the trace of the encrypted file.


Encrypted capture trace, see post request.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...

```
No.  Time       Source          Destination     Protocol Length Info
   1 0.000000   192.168.43.5    192.168.43.30   TCP        66 40851 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
   2 0.006309   192.168.43.30   192.168.43.5    TCP        66 80 → 40851 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128
   3 0.006375   192.168.43.5    192.168.43.30   TCP        54 40851 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
   4 0.006584   192.168.43.5    192.168.43.30   HTTP      745 PUT /nextcloud/ocs/v2.php/apps/end_to_end_encryption/api/v1/encrypted/563 HTTP/1.1
   5 0.009185   192.168.43.30   192.168.43.5    TCP        54 80 → 40851 [ACK] Seq=1 Ack=692 Win=30592 Len=0
   6 0.388582   192.168.43.30   192.168.43.5    HTTP/X…   805 HTTP/1.1 200 OK
   7 0.390163   192.168.43.5    192.168.43.30   HTTP      821 POST /nextcloud/ocs/v2.php/apps/end_to_end_encryption/api/v1/lock/563 HTTP/1.1  (application/x-www-form-urlencoded)
   8 0.392621   192.168.43.30   192.168.43.5    TCP        54 80 → 40851 [ACK] Seq=752 Ack=1459 Win=32128 Len=0
   9 0.707610   192.168.43.30   192.168.43.5    HTTP      790 HTTP/1.1 200 OK  (application/json)
  10 0.710590   192.168.43.5    192.168.43.30   TCP       828 40851 → 80 [PSH, ACK] Seq=1459 Ack=1488 Win=65536 Len=774 [TCP segment of a reassembled PDU]
  11 0.712840   192.168.43.30   192.168.43.5    TCP        54 80 → 40851 [ACK] Seq=1488 Ack=2233 Win=33664 Len=0
  12 0.712860   192.168.43.5    192.168.43.30   HTTP      749 POST /nextcloud/ocs/v2.php/apps/end_to_end_encryption/api/v1/meta-data/563?format=json HTTP/1.1  (application/x-www-form-urlen…
```

```
> Frame 12: 749 bytes on wire (5992 bits), 749 bytes captured (5992 bits) on interface 0
> Ethernet II, Src: IntelCor_df:05:75 (5c:e0:c5:df:05:75), Dst: Raspberr_ca:97:7d (b8:27:eb:ca:97:7d)
> Internet Protocol Version 4, Src: 192.168.43.5, Dst: 192.168.43.30
> Transmission Control Protocol, Src Port: 40851, Dst Port: 80, Seq: 2233, Ack: 1488, Len: 695
> [2 Reassembled TCP Segments (1469 bytes): #10(774), #12(695)]
> Hypertext Transfer Protocol
> HTML Form URL Encoded: application/x-www-form-urlencoded
```

```
0000  b8 27 eb ca 97 7d 5c e0  c5 df 05 75 08 00 45 00   .'...}\. ...u..E.
0010  02 df 77 24 40 00 80 06  a9 80 c0 a8 2b 05 c0 a8   ..w$@... ....+...
0020  2b 1e 9f 93 00 50 97 ec  38 54 29 dd 86 62 50 18   +....P.. 8T)..bP.
0030  01 00 34 80 00 00 6d 65  74 61 44 61 74 61 3d 25   ..4...me taData=%
0040  37 42 25 30 41 25 32 30  25 32 30 25 32 30 25 32   7B%0A%20 %20%20%2
0050  30 25 32 32 66 69 6c 65  73 25 32 32 25 33 41 25   0%22file s%22%3A%
0060  32 30 25 37 42 25 30 41  25 32 30 25 32 30 25 32   20%7B%0A %20%20%2
0070  30 25 32 30 25 37 44 25  32 43 25 30 41 25 32 30   0%20%7D% 2C%0A%20
0080  25 32 30 25 32 30 25 32  30 25 32 32 6d 65 74 61   %20%20%2 0%22meta
0090  64 61 74 61 25 32 32 25  33 41 25 32 30 25 37 42   data%22% 3A%20%7B
00a0  25 30 41 25 32 30 25 32  30 25 32 30 25 32 30 25   %0A%20%2 0%20%20%
00b0  32 30 25 32 30 25 32 30  25 32 30 25 32 32 6d 65   20%20%20 %20%22me
00c0  74 61 64 61 74 61 4b 65  79 73 25 32 32 25 33 41   tadataKe ys%22%3A
```

Frame (749 bytes) | Reassembled TCP (1469 bytes)

Encrypted capture trace, see post request | Packets: 38 · Displayed: 38 (100.0%) · Load time: 0:0.1 | Profile: Default

Figure 28. End-to-end encrypted capture trace.

8.4 Evaluation

PERFORMANCE

The NextCloud server is working as expected. The environment itself is considered to be highly secured and working properly. The tested file uploaded to the server does not increase much in size and can be synced seamlessly to another client device. However, only pre-released desktop clients have been tested since the pre-released mobile ones have not been available.

However, there are some bugs in the system that need to be fixed such as server crashing when the authentication's password is not in correct form, the tooltip hides the password instead of showing the password when clicking on "Show password", JavaScript errors while upgrading, etc. These small bugs are available of this time of writing the thesis. NextCloud is in its strongest and fastest development stage, bug fixing and upgrading can be stably released at any time in the near future.

According to the speed test, uploading and downloading inside the local network is faster than outside the local network, even though the local network's speed is much slower than the mobile hotspot's speed. Therefore, the performance of the private cloud solution is better within the local network.

THE ENVIRONMENT

According to the network setup, the environment and the data are considered to be in a highly secured and hidden place. If one would like to access to the NextCloud Server within the local network, the knowledge of Raspberry Pi's address is needed to get the access to the NextCloud server. More importantly, the Raspberry Pi must be up and running. After that, admin credentials or user credentials have to be provided to get access to the server.

If one would like to access to the NextCloud Server from outside of the local network, the public IP address of the router is needed. However, even though the server address has been port-forwarded through router, the public IP address need to be added manually to the server's trusted domain in order to be granted the authorized access. Similar to accessing within the local network, admin/ user credentials must be provided. Furthermore, the authentication can be strengthened by NextCloud's provided module, which is two-factor authentication. Additionally, even the server does not have the right to access the end-to-end encrypted data itself.

SECURITY LAYERS

There are three security layers which can be configured for the network:

- Layer 1: Filtering at the router through virtual servers
- Layer 2: Filtering at the Raspberry Pi through iptables
- Layer 3: Filtering on the NextCloud service

In the first layer, it is possible to configure the router so that only packages on port 443 and with the ip address of the Raspberry Pi be forwarded into the LAN. In the second layer, the iptables (firewall of the Raspberry Pi) can be configured on the Raspberry Pi to fully drop any incoming packages other than HTTPs-based traffic. As soon as the packages reach the NextCloud service, the service provides different layers and modules for further security. These layers and modules consits of blacklist incase the server gets brute force attacked, user’s password enforcement policy, etc.

8.5 Further development

OTHER MODULES

Beside end-to-end encryption, NextCloud private cloud server offers server-side encryption as default encryption module, and many other modules that strengthen cloud security (e.g. Two-factor authentication, auditing/logging, file access control, full-text search, etc.) (NextCloud 2018a).

OTHER PLATFORMS

There are different deployment recommendations depend on particular needs and IT infrastructure since the LAMP stack (which consists of the Linux OS, Apache HTTP Server, PHP and MySQL relational database management system) and NextCloud itself are highly configurable. The recommended scenario for small workgroups (up to 150 users) is explained as follow. The recommended system comprises of one machine (at least 2 CPU cores with 16GB RAM and local storage if needed) that runs the application server, web server, database server as well as local storage (NextCloud 2018a).

Web server	: Apache 2.4
Hypertext Pre-processor	: PHP 7.0
Database	: MariaDB, MySQL or PostgreSQL
Operating system	: Linux (either Red Har Enterprise Linux 7 or Ubuntu 16.04)

STORAGE PLAN

Since NextCloud provide unlimited storage, the cloud storage can be extended to the fullest corresponding to connected hard-disk or micro SD card. In case of increased workload, the NextCloud's data directory can be moved onto a larger external hard drive in the same way of moving NextCloud's data folder in Appendix 3.

BACKUP PLAN

In case of server crashing, a backup plan is needed. In order to backup a NextCloud installation, there are four things that must be retained, which are: the config folder, the data folder, the theme folder and the database.

The data has to be backed up on a weekly or daily basis. This either means creating image-based backups or file-based backups. In case of an image-based backup, the entire system that is running NextCloud server is copied. On the other hand, in case of a file-based backup, only the config, data and theme folders backup are taken. These two types can be used in conjunction.

If the scenario is not a disaster, the server most likely crashed and has to be rebooted. In a worst scenario, it is always possible to have a running server within an hour of downtime as backups have been taken consistently.

9 CONCLUSION

The thesis paper consists of two main parts, the theoretical background and the empirical part. The theoretical background part delivered information on the concept of cloud computing, data security standards and issues in cloud computing, data encryption in cloud computing and the current issues with public cloud service providers and private cloud storage applications. The theoretical background is written to give readers a deeper look into the latest issues in data security in cloud computing and be familiar with the concept of end-to-end encryption. The main aim is to prove that end-to-end encryption is the key to a high-level data security in cloud computing. Many public cloud service providers and private cloud storage applications are compared with each other so that users can have a better look at the services and the prices of the services they are offering.

The empirical part consists of the implementation and testing of the private cloud solution built with NextCloud. The environment for the private cloud server is written in the implementation plan. The results of the testing phase are stated in the testing and monitoring section. The results are shown to affirm that private cloud solution can be a low-budget, integrated and secure solution that enables user to store, back-up and control their confidential data themselves. Moreover, the performance is proven to be better within the local network.

The thesis's main aim is not only giving readers a guideline to build their own private cloud server, but also giving readers information on different cloud solutions in the market. Therefore, users can choose which solutions fit their requirements and needs.

The result of the thesis is a private cloud solution built with NextCloud and it is proven in the thesis work that the solution is well functioning and secured. The practical part of the thesis can be used as the guideline for building one's own private cloud server at home. The significance of the thesis work is proving that building a private cloud server enables users to have more control over their own hardware infrastructure, experience best speed in device synchronization, better privacy assurance, and lower in cost of cloud services.

From the author's perspective, it would be better to store general documents on public cloud for easy sharing and collaborating (e.g. team-work assignment, etc.). However, when it comes to confidential data (e.g. health documents, bank statements, etc.), a more secure solution is needed.

In the thesis candidate own’s perspective, the thesis work has not only been a theoretical but also practical and innovative way to learn about security in cloud computing and especially encryption methodologies and techniques. During the period the thesis is being carried on, there has been many problems with NextCloud since there are still many bugs. In general, the thesis candidate has learnt how to utilize RaspberryPi in building a private cloud server, different encryption methodologies and techniques, enable SSL, port-forwarding, network traffic monitoring using WireShark, and upload and download speed testing. Since the thesis candidate’s major is Software Engineering, these topics in Cloud Computing are not only challenging but also interesting.

References

Books and publications

Badger, L. Grance, T. Patt-Corner, R. & Voas, J. 2012. Cloud Computing Synopsis and Recommendations. Recommendations of the National Institute of Standards and Technology, 800, 146, pp. 8.7–8.9.

Barker, E. 2016. Guideline for Using Cryptographic Standards in the Federal Government: Cryptographic Mechanisms, 800, 175B, pp. 19–36.

Bellare, M., Canetti, R. & Krawczyk, H. 1996. Keying Hash Functions for Message Authentication. Advances in Cryptology – Crypto 96, 1109, pp 3–15.

Camenisch, J., Fischer-Hubner, S. & Rannenberg, K. 2011. Privacy and Entity Management for Life. Springer. Berlin, pp. 185–186.

Chen, L. 2009. Recommendation for Key Derivation Using Pseudorandom Functions (Revised). NIST Special Publication 800-108, pp. 2–21.

Das, S. Hossain, M. Sardar, M. Biswas, R. Nath, P. 2014. Performance Analysis of Client-Side Encryption Tools. International Journal of Advanced Computer Research, 4, 16, 3, pp 888–897.

Heiser, J. 21 August 2014. Research Vice President. Understanding and Controlling the Risks of Cloud Computing. Gartner. Web-based seminar presentation.

Huang, K. Chiu, J. Shen, S. 2013. A Novel Structure with Dynamic Operating Mode for Symmetric-Key Block Ciphers. International Journal of Network Security & Its Applications, 5, 1, pp. 17–36.

Jakimoski, K. 2016. Security Techniques for Data Protection in Cloud Computing. International Journal of Grid and Distributed Computing, 9, 1, pp. 49–56.

Joux, A. 2004. Multicollisions in Iterated Hash Functions. Application to Cascaded Constructions. Advances in Cryptology – CRYPTO 2004, LNCS, 3152, pp. 306–316.

Mell, P. & Grance, T. 2011. The NIST Definition of Cloud Computing. Recommendations of the National Institute of Standards and Technology, 800, 145, pp. 2–3.

Moriarty, K. Kaliski, B. Rusch, A. 2017. PKCS #5: Password-Based Cryptography Specification Version 2.1. Internet Engineering Task Force, 2070-1721, pp. 2–13.

Netto, M. Calheiros, R. Rodrigues, E. Cunha, R. & Buyya, R. 2018. HPC Cloud for Scientific and Business Applications: Taxonomy, Vision, and Research Challenges. ACM Computing Surveys (CSUR), 51,1, pp. 1–29.

Ngo, T. 2017. Data Security and Privacy in Cloud Computing. Research Seminar Final Report, Haaga-Helia University of Applied Sciences. Accessed 10 September 2018.

NIST 2001. Announcing the Advanced Encryption Standard (AES). Federal Information Processing Standards Publication 197.

Sen, A. & Tiwari, P. 2017. Security Issues and Solutions in Cloud Computing. IOSR Journal of Computer Engineering (IOSR-JCE), 19, 2, pp. 70–71.

Turner, S. Chen, L. 2011. Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms, Internet Engineering Task Force, 2070-1721, pp. 1–7.

Online sources

Apple 2017. iCloud security overview. URL: https://support.apple.com/en-us/HT202303. Accessed 23 April 2018.

Boxcryptor 2018. AES and RSA Encryption. URL: https://www.boxcryptor.com/en/encryption/. Accessed 28 April 2018.

Chang, L. 2017. Encryption's role in GDPR compliance and cloud data security. URL: https://www.itproportal.com/features/encryptions-role-in-gdpr-compliance-and-cloud-data-security/. Accessed 22 March 2018.

Cloud Security Alliance 2017. The Treacherous 12 – Top Threats to Cloud Computing + Industry Insights. Seattle. URL: https://cloudsecurityalliance.org/download/top-threats-cloud-computing-plus-industry-insights/. Accessed 22 March 2018.

Cloud Standards Customer Council (CSCC). 2016. Cloud Security Standards: What to Expect & What to Negotiate. Version 2.0. URL: http://www.cloud-council.org/deliverables/CSCC-Cloud-Security-Standards-What-to-Expect-and-What-to-Negotiate.pdf. Accessed 02 April 2018.

Cloudfogger 2016. Cloudfogger project has been stopped. URL: https://www.cloudfogger.com/. Accessed 23 April 2018.

Danova, T. 2014. Most People Are Still Confused About Cloud Storage, and No One Service Is Winning The Race To Educate And Acquire Users. URL: http://www.businessinsider.com/people-use-the-cloud-and-dont-even-realize-it-2014-7?r=US&IR=T&IR=T. Accessed 21 March 2018.

Deshmukh, S. 2016. Importance of cloud computing. URL: https://www.esds.co.in/blog/importance-of-cloud-computing/#sthash.AFPV9cTH.dpbs. Accessed 31 March 2018.

Dinu, D. 2017. The Password Hash Argon2, Winner of PHC. URL: https://github.com/P-H-C/phc-winner-argon2. Accessed 20 May 2018.

Dropbox 2018. Under the hood: Architecture overview. URL: https://www.dropbox.com/business/trust/security/architecture. Accessed 23 April 2018.

Dutton, J. 2017. How ISO 27001 can help to achieve GDPR compliance. URL: https://www.itgovernance.co.uk/blog/how-iso-27001-can-help-to-achieve-gdpr-compliance/. Accessed 19 May 2018.

Fu, A. 2017. 7 Different Types of Cloud Computing Structures. URL: https://www.uniprint.net/en/7-types-cloud-computing-structures/. Accessed 22 April 2018.

Google Cloud 2018a. Data Encryption Options. URL: https://cloud.google.com/storage/docs/encryption/. Accessed 23 April 2018.

Google Cloud 2018b. Client-Side Encryption Keys. URL: https://cloud.google.com/storage/docs/encryption/client-side-keys. Accessed 23 April 2018.

Google Trends 2018. Compare. URL: https://trends.google.com/trends/explore?date=2016-04-11%202018-05-11&q=ownCloud,Nextcloud,Seafile,Pydio. Accessed 11 May 2018.

Goldman, J. 2015. Bitdefender Acknowledges Data Breach. URL: https://www.esecurityplanet.com/network-security/bitdefender-acknowledges-data-breach.html. Accessed 22 March 2018.

Gueron, S. 2013. AES-GCM for Efficient Authenticated Encryption – Ending the Reign of HMAC-SHA-1? URL: https://crypto.stanford.edu/RealWorldCrypto/slides/gueron.pdf. Accessed 20 May 2018.

HackerOne 2017. NextCloud. URL: https://hackerone.com/nextcloud. Accessed 11 May 2018.

Hamdaqa, M. & Tahvildari, L. 2012. Cloud Computing Uncovered: A Research Landscape. Elsevier Press. URL: http://www.stargroup.uwaterloo.ca/~mhamdaqa/publications/Cloud_Computing_Uncovered.pdf. Accessed 02 April 2018.

ISO 27001 Security. 2018. About the ISO27k standards. URL: http://www.iso27001security.com/html/iso27000.html. Accessed 26 April 2018.

Mell, P. & Grance, T. 7 October 2009. Project Lead & Project Manager. Effectively and Securely Using the Cloud Computing Paradigm. National Institute of Standards and Technology (NIST) Information Technology Laboratory. Seminar presentation.

Microsoft 2018. HMACSHA1 Class. URL: https://msdn.microsoft.com/en-us/library/system.security.cryptography.hmacsha1(v=vs.110).aspx. Accessed 20 May 2018.

NextCloud 2017a. End-to-end encryption design. NextCloud's Whitepapers, pp. 1–17.

NextCloud 2017b. Bring enterprise data back under control with NextCloud. URL: https://nextcloud.com/blog/bring-enterprise-data-back-under-control-with-nextcloud/. Accessed 19 May 2018.

NextCloud 2018a. URL: https://apps.nextcloud.com/. Accessed 16 April 2018.

NextCloud 2018b. German Federal Administration relies on NextCloud as a secure file exchange solution. URL: https://nextcloud.com/blog/german-federal-administration-relies-on-nextcloud-as-a-secure-file-exchange-solution/. Accessed 11 May 2018.

NIST 2018a. Block Cipher Techniques. URL: https://csrc.nist.gov/projects/block-cipher-techniques. Accessed 19 May 2018.

NIST 2018b. Hash Functions. URL: https://csrc.nist.gov/Projects/Hash-Functions. URL: https://csrc.nist.gov/Projects/Hash-Functions. Accessed 20 May 2018.

ownCloud 2018. Pricing. URL: https://owncloud.com/pricing/. Accessed 07 May 2018.

PKWARE. 2018. Client-side Encryption vs. End-to-End Encryption: What's the Difference? URL: https://www.pkware.com/blog/client-side-encryption-vs-end-to-end-encryption-what-s-the-difference. Accessed 03 April 2018.

Pydio 2018. Encryption at rest. URL: https://pydio.com/en/docs/kb/security/encryption-rest. Accessed 11 May 2018.

Rode, A. 2017. Server-side Encryption – Securing data at rest. NextCloud's Whitepapers, pp. 1–6.

Rouse, M. 2017a. Definition of Multi-cloud strategy. URL: http://searchcloudcomputing.techtarget.com/definition/multi-cloud-strategy. Accessed 02 April 2018.

Rouse, M. 2017b. Definition of Public Cloud. URL: http://searchcloudcomputing.techtarget.com/definition/public-cloud. Accessed 02 April 2018.

Rouse, M. 2017c. Definition of Private Cloud (Internal Cloud or Corporate Cloud). URL: http://searchcloudcomputing.techtarget.com/definition/private-cloud. Accessed 02 April 2017.

Salcedo, H. 2014. Open Source ownCloud Offers Client-side Encryption. URL: https://psg.hitachi-solutions.com/credeon/blog/open-source-owncloud-offers-client-side-encryption. Accessed 22 March 2018.

Seafile 2018. Deploy Seafile on Your Own Server. URL: https://www.seafile.com/en/product/private_server/. Accessed 11 May 2018.

Tietz, S. 2013. Cloud encryption – Client-side vs Server-side. Stackfield Blog. URL: https://www.stackfield.com/blog/cloud-encryption---client-side-vs-server-side-1. Accessed 03 April 2018.

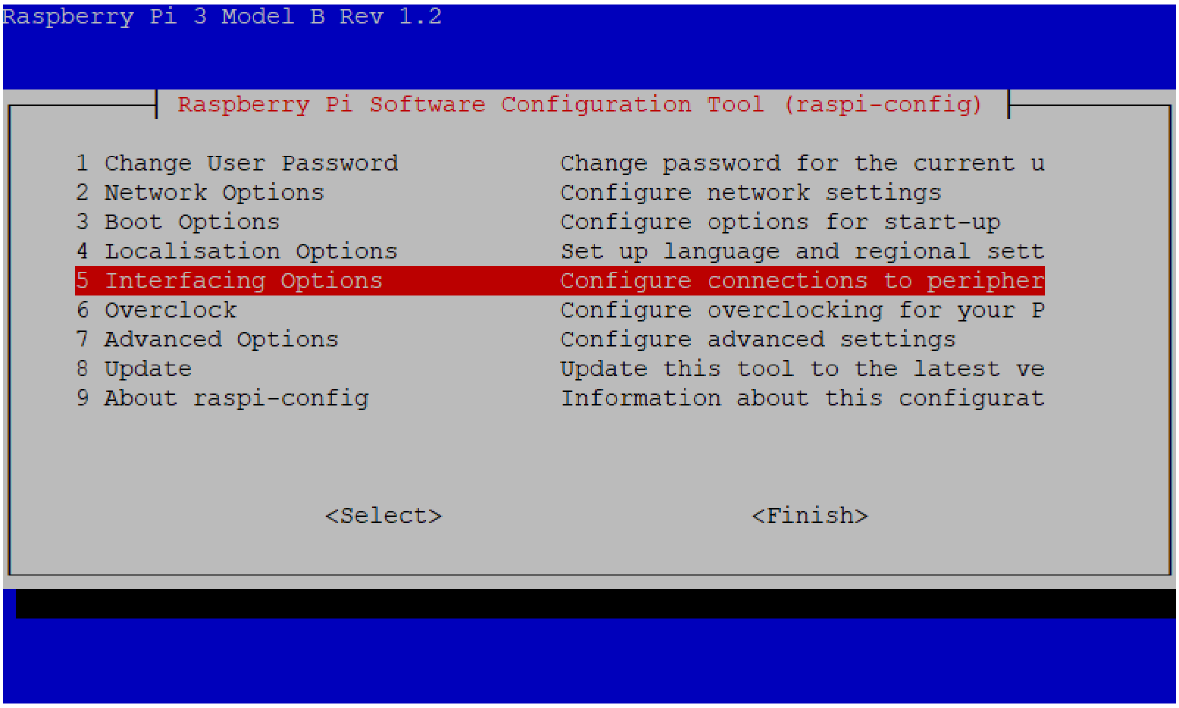
Tutorialspoint 2018. Cloud Computing Architecture. URL: https://www.tutorialspoint.com/cloud_computing/cloud_computing_architecture.htm. Accessed 07 May 2018.

Winkler, V. 2011. Cloud Computing: Virtual Cloud Security Concerns. URL: https://technet.microsoft.com/en-us/library/hh641415.aspx. Accessed 02 April 2018.

Zafer, T. 2016. Why Client-side Encryption is the Next Best Idea in Cloud-Based Data Security. http://www.infosectoday.com/Articles/Client-Side_Encryption.htm. Accessed 03 April 2018.

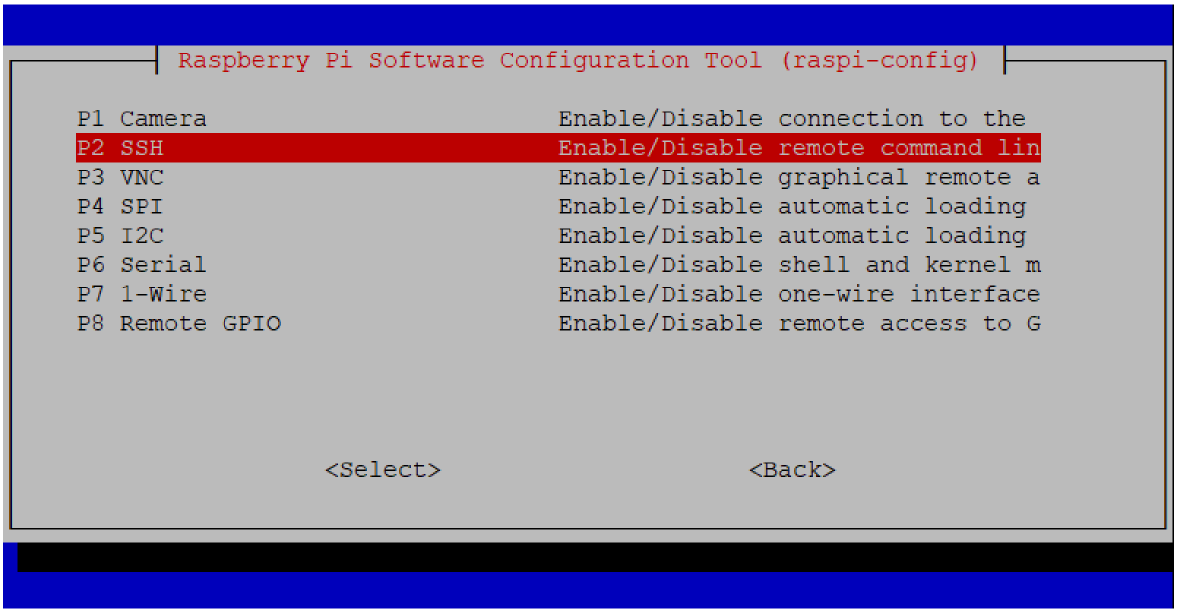
Appendices

Appendix 1: Connecting to the Raspberry Pi 3



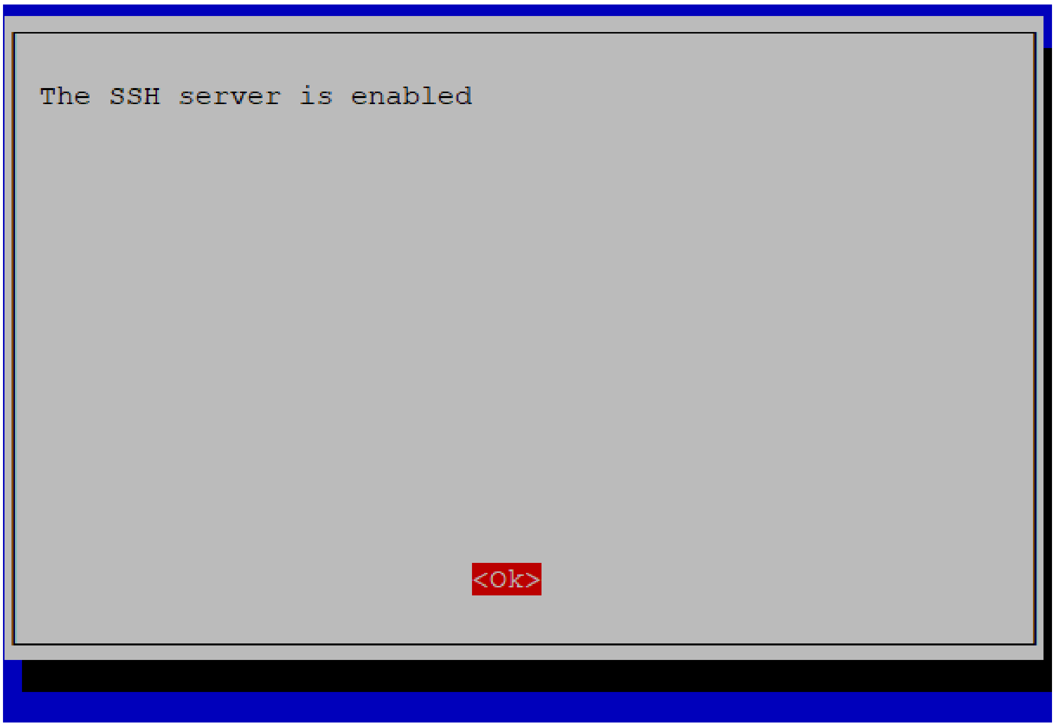
```
Raspberry Pi 3 Model B Rev 1.2

         Raspberry Pi Software Configuration Tool (raspi-config)

    1 Change User Password          Change password for the current u
    2 Network Options               Configure network settings
    3 Boot Options                  Configure options for start-up
    4 Localisation Options          Set up language and regional sett
    5 Interfacing Options           Configure connections to peripher
    6 Overclock                     Configure overclocking for your P
    7 Advanced Options              Configure advanced settings
    8 Update                        Update this tool to the latest ve
    9 About raspi-config            Information about this configurat


                   <Select>                     <Finish>
```

Image 1. Configure connections to Raspberry Pi 3



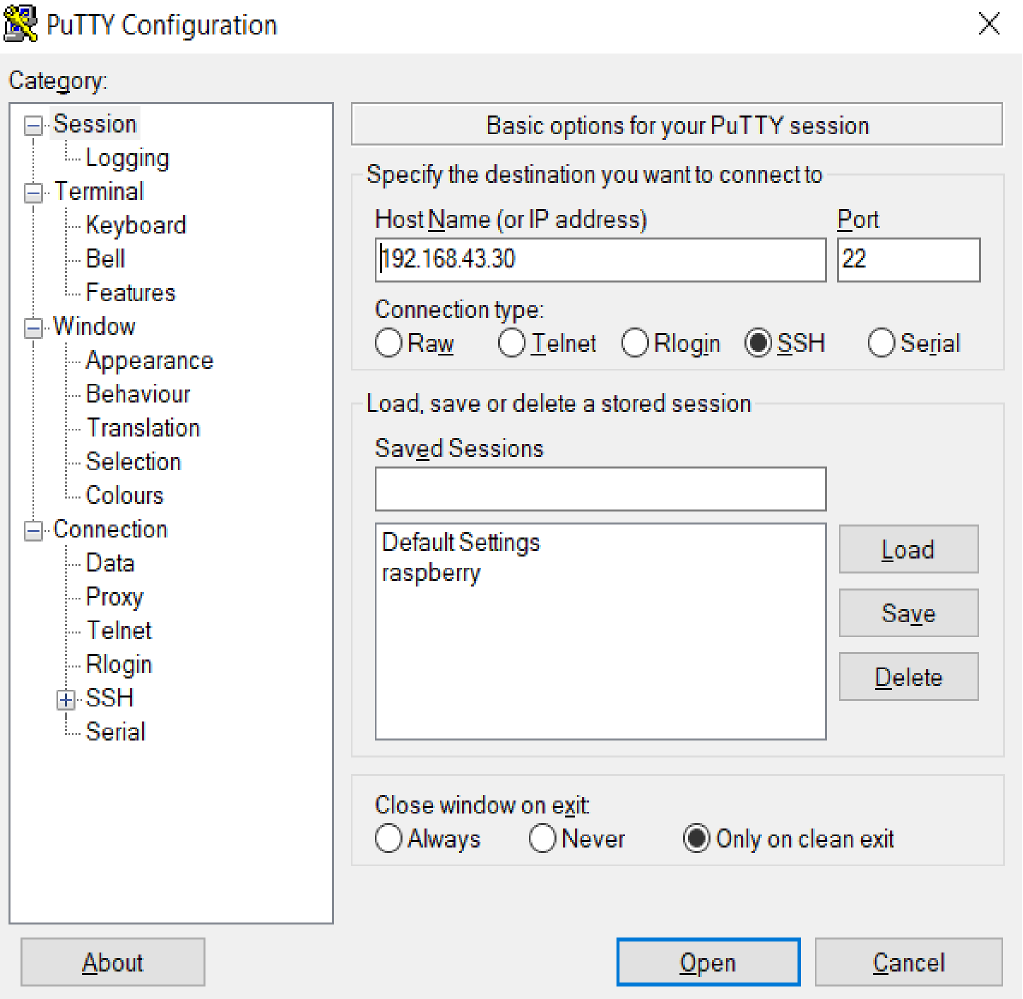
```
         Raspberry Pi Software Configuration Tool (raspi-config)

    P1 Camera                       Enable/Disable connection to the
    P2 SSH                          Enable/Disable remote command lin
    P3 VNC                          Enable/Disable graphical remote a
    P4 SPI                          Enable/Disable automatic loading
    P5 I2C                          Enable/Disable automatic loading
    P6 Serial                       Enable/Disable shell and kernel m
    P7 1-Wire                       Enable/Disable one-wire interface
    P8 Remote GPIO                  Enable/Disable remote access to G


                   <Select>                     <Back>
```

Image 2. Enable SSH in Raspberry Pi’s configuration tool



```
The SSH server is enabled
```

Image 3. SSH is enabled in Raspberry Pi’s configuration tool



Specify the destination you want to connect to

Host Name (or IP address): 192.168.43.30

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

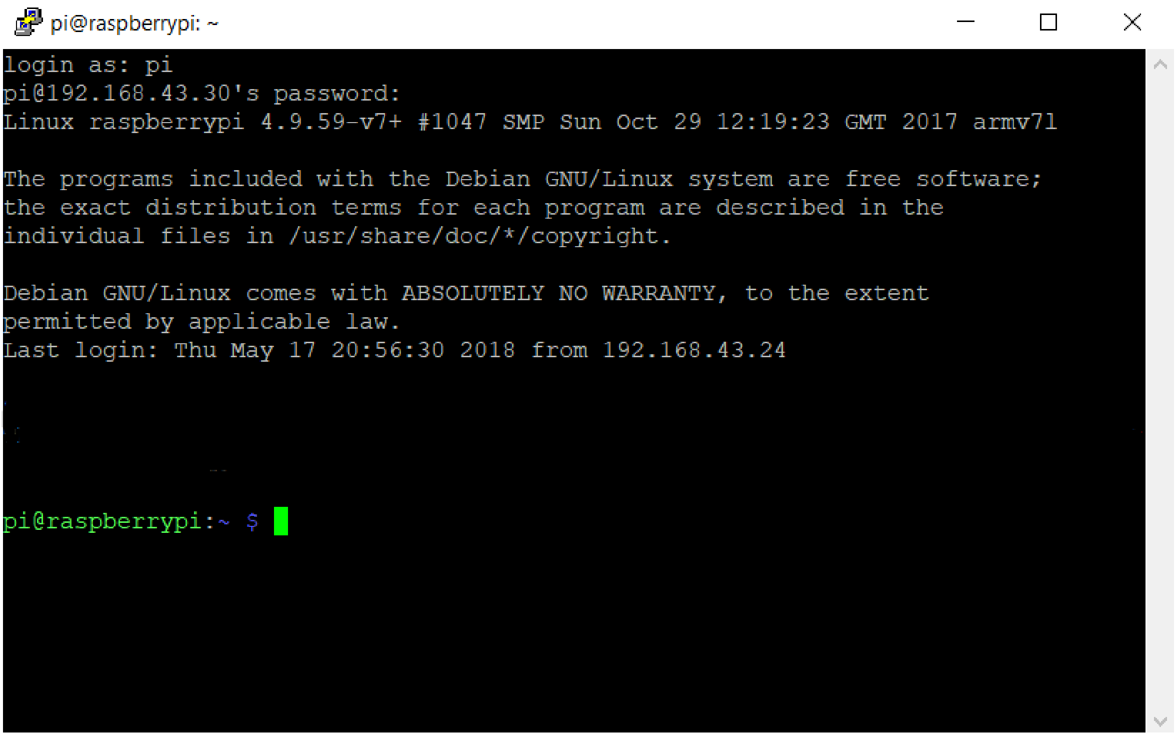
Load, save or delete a stored session

Saved Sessions:

Default Settings
raspberry

Close window on exit: ○ Always ○ Never ● Only on clean exit

Image 4. Establish connection to Raspberry Pi using PuTTY



```
login as: pi
pi@192.168.43.30's password:
Linux raspberrypi 4.9.59-v7+ #1047 SMP Sun Oct 29 12:19:23 GMT 2017 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu May 17 20:56:30 2018 from 192.168.43.24


pi@raspberrypi:~ $
```

Image 5. Connection established, and login succeeded.

Appendix 2: Pre-installation

Step 1: Update package repositories

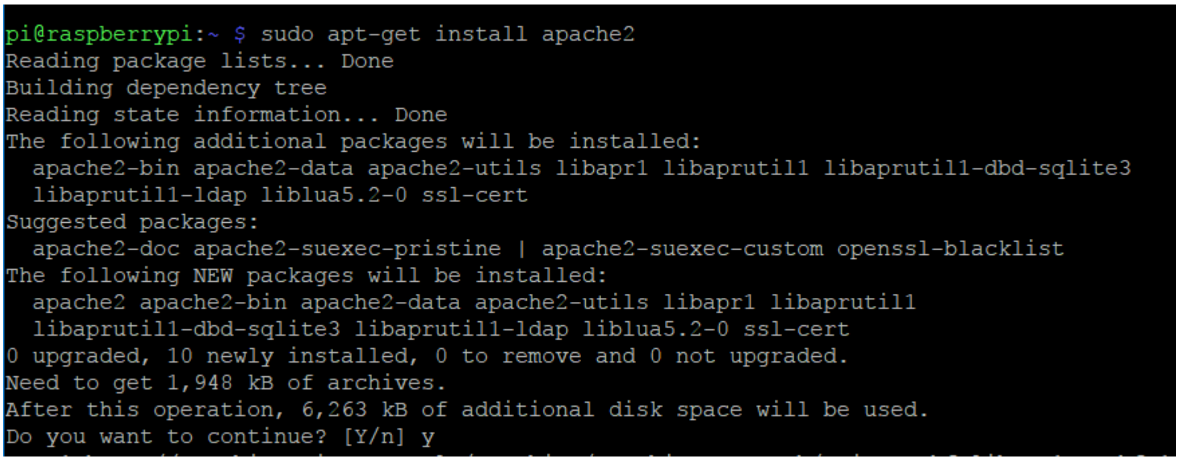
# sudo apt-get update

# sudo apt-get upgrade

Step 2: Install Apache HTTP Server

The Apache HTTP Server is the most popular, free and open-source cross-platform web server.

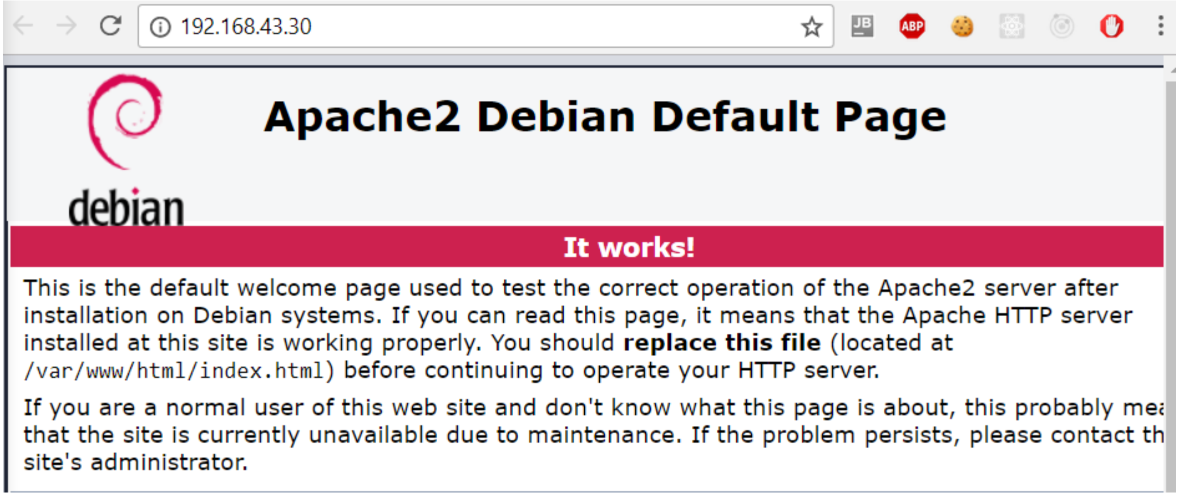
# sudo apt-get install apache2



```
pi@raspberrypi:~ $ sudo apt-get install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap liblua5.2-0 ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom openssl-blacklist
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0 ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 0 not upgraded.
Need to get 1,948 kB of archives.
After this operation, 6,263 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

Image 6. Install Apache HTTP Server

Apache2 can be checked if it is up and running by going the Raspberry Pi's address.



192.168.43.30

Apache2 Debian Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

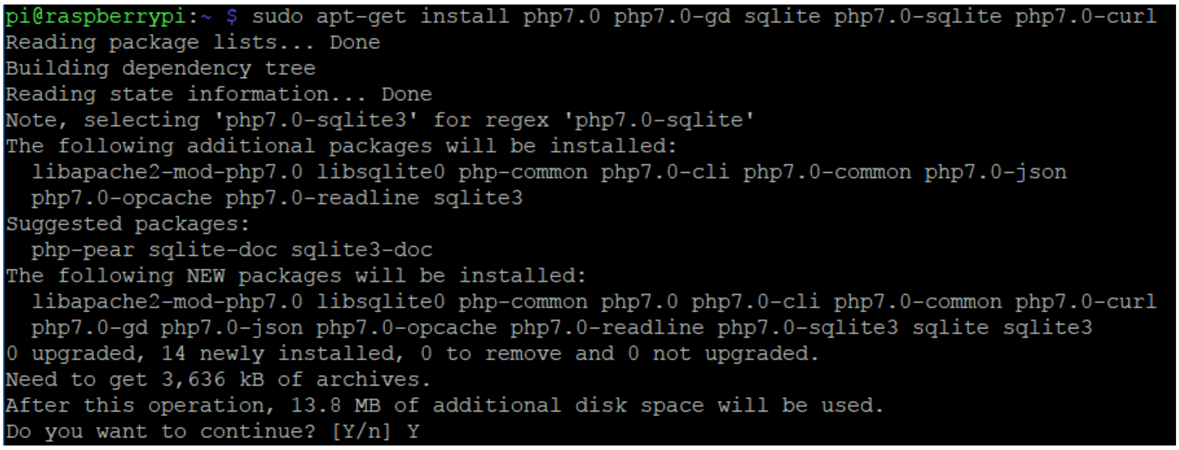
If you are a normal user of this web site and don't know what this page is about, this probably mea that the site is currently unavailable due to maintenance. If the problem persists, please contact th site's administrator.

Image 7. Apache2 install confirmation

Step 3: Install PHP and its packages

PHP (or Pre-processor) is a server-side scripting language, which is not only especially designed for web developing but also a general-purpose programming language.

# sudo apt-get install php7.0 php7.0-gd sqlite php7.0-sqlite php7.0-curl

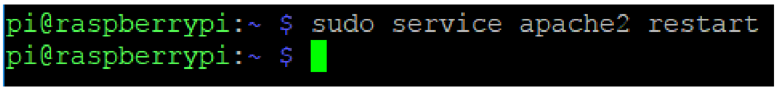


```
pi@raspberrypi:~ $ sudo apt-get install php7.0 php7.0-gd sqlite php7.0-sqlite php7.0-curl
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'php7.0-sqlite3' for regex 'php7.0-sqlite'
The following additional packages will be installed:
  libapache2-mod-php7.0 libsqlite0 php-common php7.0-cli php7.0-common php7.0-json
  php7.0-opcache php7.0-readline sqlite3
Suggested packages:
  php-pear sqlite-doc sqlite3-doc
The following NEW packages will be installed:
  libapache2-mod-php7.0 libsqlite0 php-common php7.0 php7.0-cli php7.0-common php7.0-curl
  php7.0-gd php7.0-json php7.0-opcache php7.0-readline php7.0-sqlite3 sqlite sqlite3
0 upgraded, 14 newly installed, 0 to remove and 0 not upgraded.
Need to get 3,636 kB of archives.
After this operation, 13.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

Image 8. Install PHP and PHP packages

Step 4: Restart Apache

# sudo service apache2 restart



```
pi@raspberrypi:~ $ sudo service apache2 restart
pi@raspberrypi:~ $
```

Image 9. Restart Apache

Appendix 3: NextCloud Installation

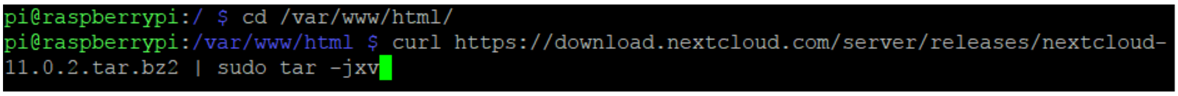
Step 1: Install NextCloud in Apache2 HTTP Server

The first thing is do is moving the working directory to the html directory

# cd /var/www/html/

After that, install NextCloud in html directory

# curl https https://download.nextcloud.com/server/releases/nextcloud-11.0.2.tar.bz2 | sudo tar -jxv



```
pi@raspberrypi:/ $ cd /var/www/html/
pi@raspberrypi:/var/www/html $ curl https://download.nextcloud.com/server/releases/nextcloud-
11.0.2.tar.bz2 | sudo tar -jxv
```

Image 10. Install NextCloud 11.0.2

Step 2: Setting directory and access rights

Change the working directory to the unzipped folder by this command

# cd var/www/html/nextcloud

The following command is to create a new data folder for NextCloud to operate in

# sudo mkdir -p /var/www/html/nextcloud/data

Granting access for correct user and group control over the data directory by the command below

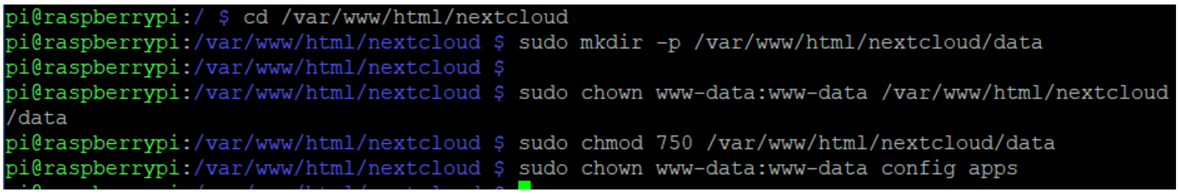
# sudo chown www-data:www-data /var/www/html/nextcloud/data

Giving right permissions

# sudo chmod 750 /var/www/html/nextcloud/data

Giving www-data group control over the configuration and applications folder

# sudo chown www-data:www-data config apps

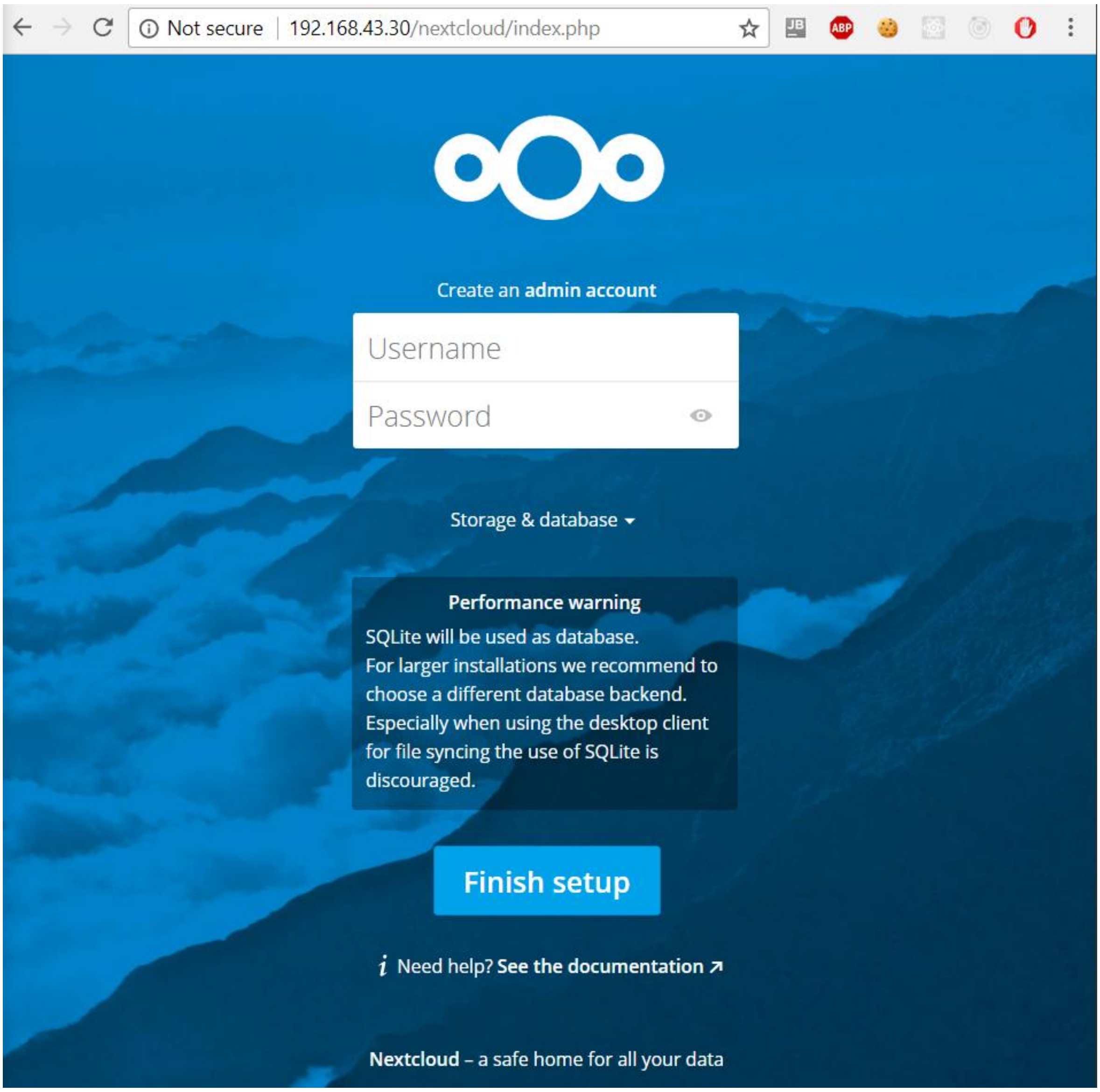


```
pi@raspberrypi:/ $ cd /var/www/html/nextcloud
pi@raspberrypi:/var/www/html/nextcloud $ sudo mkdir -p /var/www/html/nextcloud/data
pi@raspberrypi:/var/www/html/nextcloud $
pi@raspberrypi:/var/www/html/nextcloud $ sudo chown www-data:www-data /var/www/html/nextcloud
/data
pi@raspberrypi:/var/www/html/nextcloud $ sudo chmod 750 /var/www/html/nextcloud/data
pi@raspberrypi:/var/www/html/nextcloud $ sudo chown www-data:www-data config apps
pi@raspberrypi:/var/www/html/nextcloud $
```

Image 11. Setting directory and access rights

Step 3. Run NextCloud

Go to the Raspberry Pi’s address and add /nextcloud


Not secure
192.168.43.30/nextcloud/index.php
Create an admin account
Username
Password
Storage & database
Finish setup
Need help? See the documentation ↗
Nextcloud – a safe home for all your data

Performance warning

SQLite will be used as database.
For larger installations we recommend to choose a different database backend.
Especially when using the desktop client for file syncing the use of SQLite is discouraged.

Image 12. NextCloud up and running

Appendix 4: Other Setups

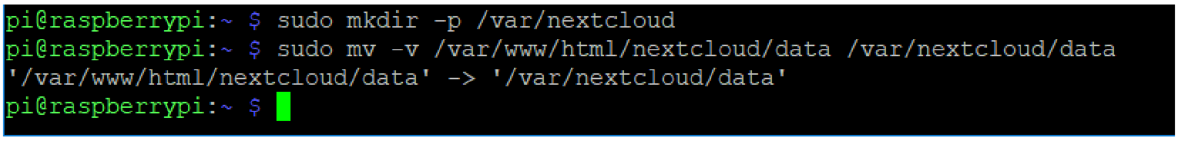
Step 1. Moving data directory on the server side

Create a new folder

```
# sudo mkdir -p /var/nextcloud
```

Move the data directory into the new folder created

```
# sudo mv -v /var/www/html/nextcloud/data /var/nextcloud/data
```



```
pi@raspberrypi:~ $ sudo mkdir -p /var/nextcloud
pi@raspberrypi:~ $ sudo mv -v /var/www/html/nextcloud/data /var/nextcloud/data
'/var/www/html/nextcloud/data' -> '/var/nextcloud/data'
pi@raspberrypi:~ $
```

Image 13. Moving data folder

Change to the configuration directory

# cd /var/www/html/nextcloud/config

Copy and make a backup of the file

# sudo cp -p config.php config.php.bk

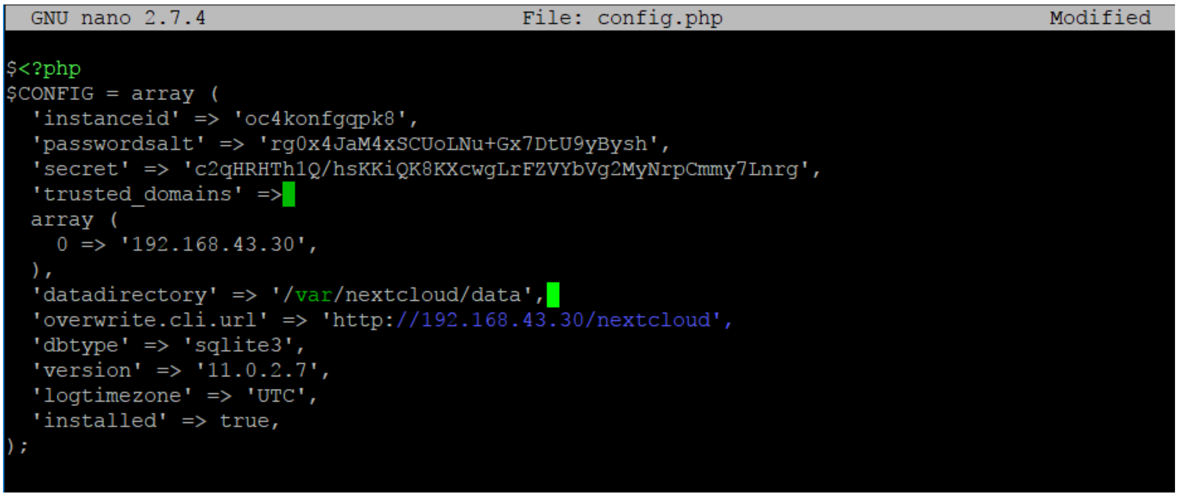
Open and edit the configuration file

# sudo nano config.php

Modify data directory config to point to the new directory

Used to be ‘datadirectory’ => ‘/var/www/html/nextcloud/data’

Changed to ‘datadirectory’ => ‘/var/nextcloud/data’



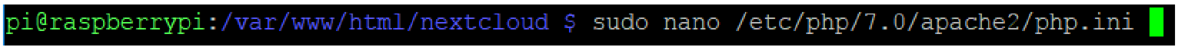
```
  GNU nano 2.7.4                          File: config.php                          Modified

$<?php
$CONFIG = array (
  'instanceid' => 'oc4konfgqpk8',
  'passwordsalt' => 'rg0x4JaM4xSCUoLNu+Gx7DtU9yBysh',
  'secret' => 'c2qHRHTh1Q/hsKKiQK8KXcwgLrFZVYbVg2MyNrpCmmy7Lnrg',
  'trusted_domains' =>
  array (
    0 => '192.168.43.30',
  ),
  'datadirectory' => '/var/nextcloud/data',
  'overwrite.cli.url' => 'http://192.168.43.30/nextcloud',
  'dbtype' => 'sqlite3',
  'version' => '11.0.2.7',
  'logtimezone' => 'UTC',
  'installed' => true,
);
```

Image 14. Change location in config file

Step 2. Increasing max upload size

Edit the config file

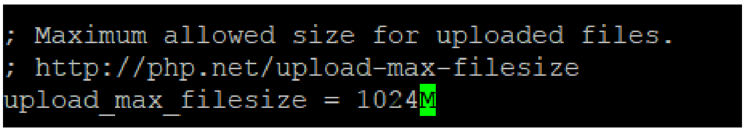
# sudo nano /etc/php5/apache2/php.ini



```
pi@raspberrypi:/var/www/html/nextcloud $ sudo nano /etc/php/7.0/apache2/php.ini
```

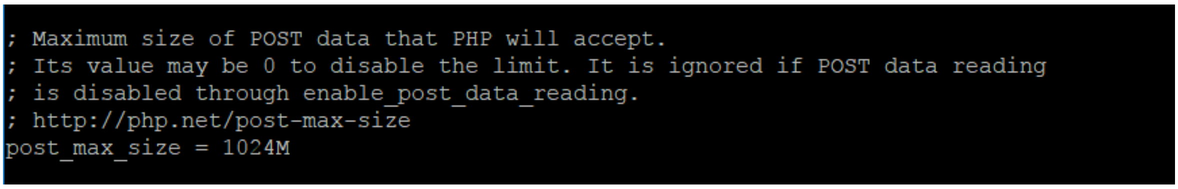
Image 15. Edit the config file

Increase the post max size and upload max size by editing upload_max_filesize and post_max_size to 1024M



```
; Maximum allowed size for uploaded files.
; http://php.net/upload-max-filesize
upload_max_filesize = 1024M
```

Image 16. Increase upload max file size



```
; Maximum size of POST data that PHP will accept.
; Its value may be 0 to disable the limit. It is ignored if POST data reading
; is disabled through enable_post_data_reading.
; http://php.net/post-max-size
post_max_size = 1024M
```

Image 17. Increase post max size

Restart Apache for it to load the config file

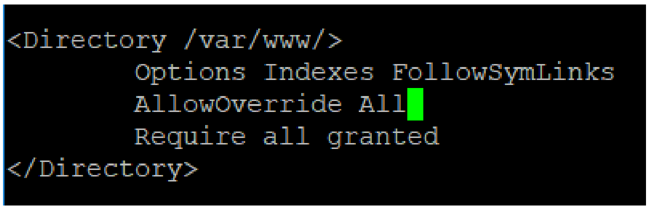
# sudo service apache2 restart

Step 3. Allowing the .htaccess override

Access the apache2 config file

# sudo nano /etc/apache2.conf

Allow override



```
<Directory /var/www/>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
</Directory>
```

Image 18. Allowing .htaccess override

Restart Apache for it to read the changed config file

# sudo service apache2 restart

Step 4. SSL Setup

Creating a folder to store the self-signed certificate

# sudo mkdir -p /etc/apache2/ssl

Generate the self-signed certificate using OpenSSL

reg : subcommand for X.507 CSR management (certificate signing request)

-x509 : instead of sending a certificate request, make a self-signed one

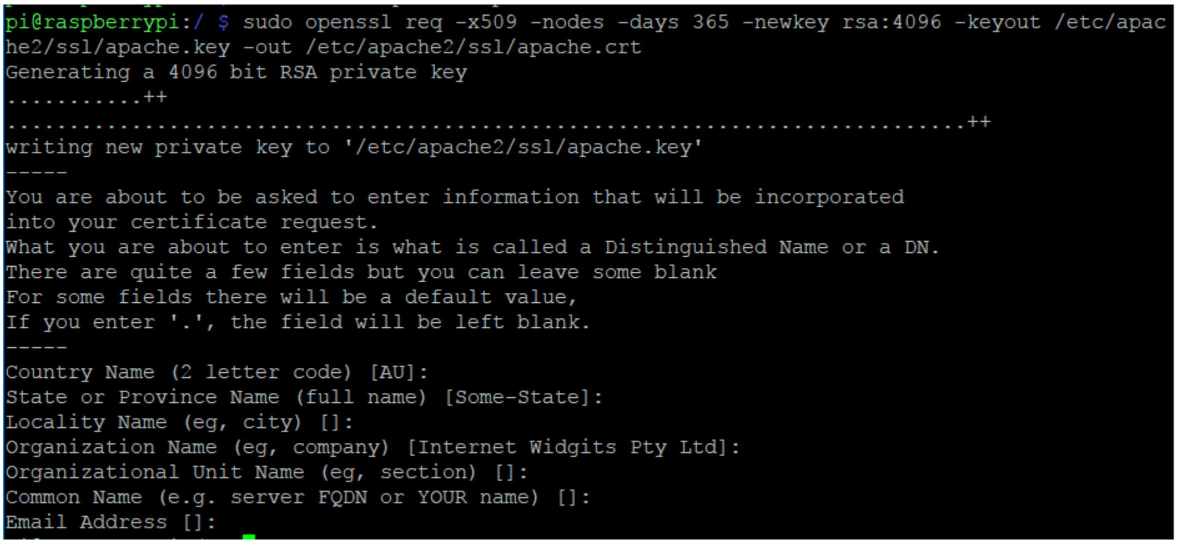
-nodes : no passphrase given and asked every time apache2 restarts

-days 365 : the number of days that the self-signed certificate remains valid for After 365 days, a new certificate needs to be generated.

-newkey rsa:4096 : create certificate request and private key (RSA key 2048 bits long)

-keyout : defines the output file for private key file

-out : defines the output file for the self-signed certificate

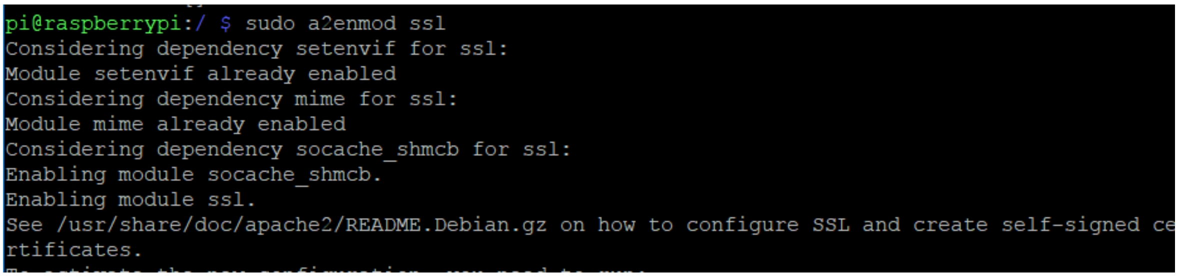


```
pi@raspberrypi:/ $ sudo openssl req -x509 -nodes -days 365 -newkey rsa:4096 -keyout /etc/apac
he2/ssl/apache.key -out /etc/apache2/ssl/apache.crt
Generating a 4096 bit RSA private key
..........++
............................................................................++
writing new private key to '/etc/apache2/ssl/apache.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:
```

Image 19. Generate self-signed certificate

Enable SSL Module

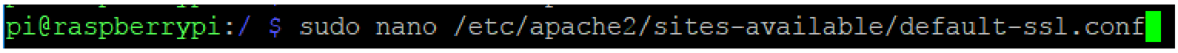
# sudo a2enmod ssl



```
pi@raspberrypi:/ $ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed ce
rtificates.
```

Image 20. Enable SSL

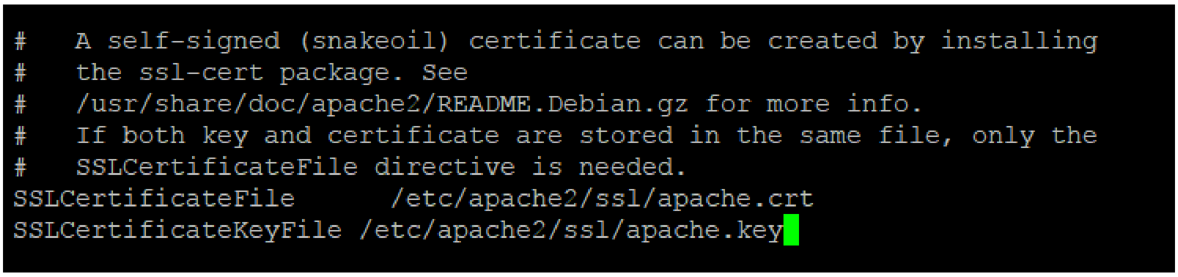
Modify default-ssl.conf so the config file will use the self-signed certificate instead of the default one



```
pi@raspberrypi:/ $ sudo nano /etc/apache2/sites-available/default-ssl.conf
```

Image 21. Modify default-ssl.conf

Changes should be made on the SSLCertificateFile and SSLCertificateKeyFile in order to point the generated certificate into the /etc/apache2/ssl/ directory.



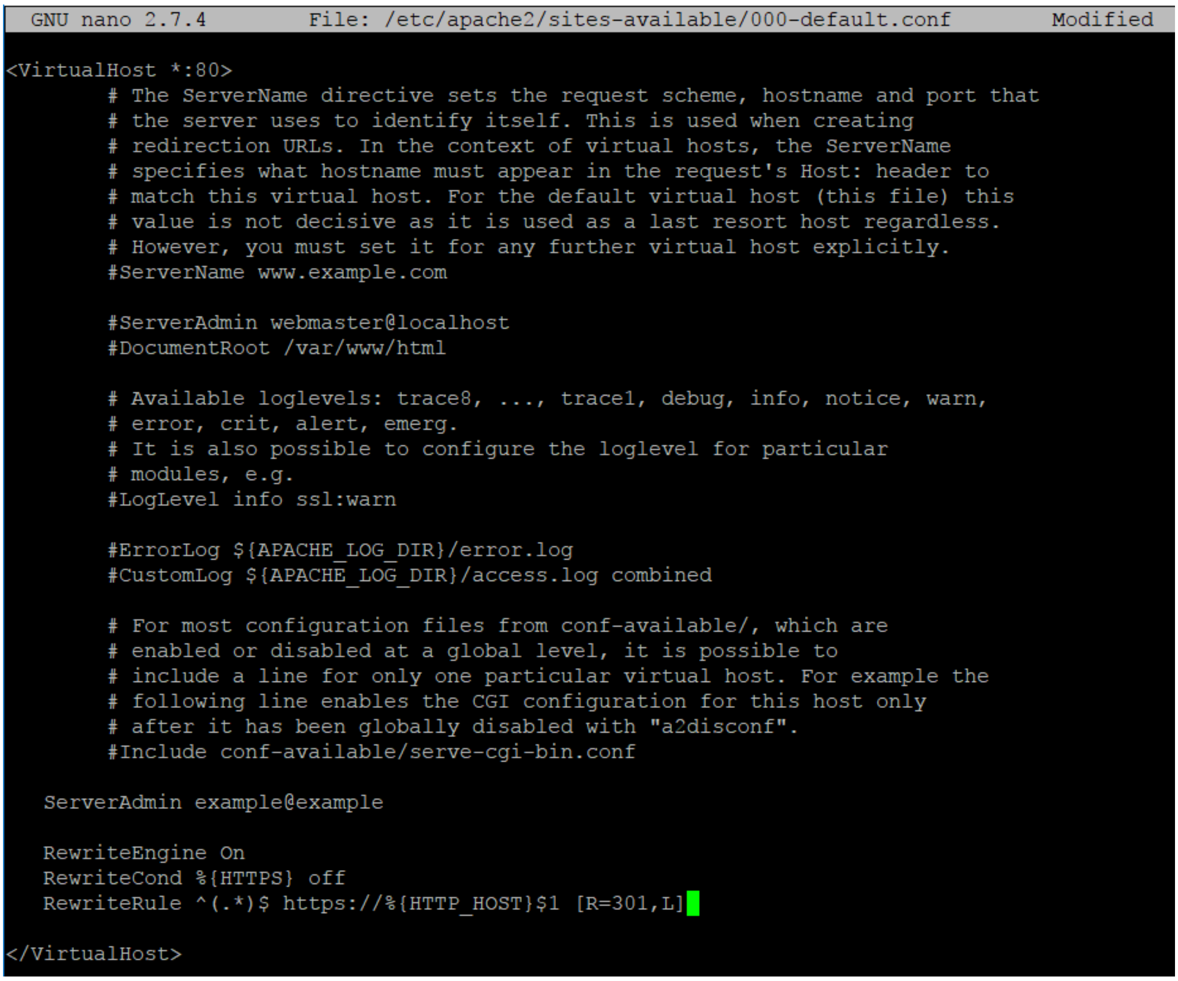
```
#   A self-signed (snakeoil) certificate can be created by installing
#   the ssl-cert package. See
#   /usr/share/doc/apache2/README.Debian.gz for more info.
#   If both key and certificate are stored in the same file, only the
#   SSLCertificateFile directive is needed.
SSLCertificateFile      /etc/apache2/ssl/apache.crt
SSLCertificateKeyFile /etc/apache2/ssl/apache.key
```

Image 22. Changes in the default-ssl.conf

Enforce SSL so there is no connection can be made over HTTP by editing the default file

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

Redirect HTTP traffic to the equivalent HTTPs as follow



```
<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        #ServerName www.example.com

        #ServerAdmin webmaster@localhost
        #DocumentRoot /var/www/html

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.
        #LogLevel info ssl:warn

        #ErrorLog ${APACHE_LOG_DIR}/error.log
        #CustomLog ${APACHE_LOG_DIR}/access.log combined

        # For most configuration files from conf-available/, which are
        # enabled or disabled at a global level, it is possible to
        # include a line for only one particular virtual host. For example the
        # following line enables the CGI configuration for this host only
        # after it has been globally disabled with "a2disconf".
        #Include conf-available/serve-cgi-bin.conf

   ServerAdmin example@example

   RewriteEngine On
   RewriteCond %{HTTPS} off
   RewriteRule ^(.*)$ https://%{HTTP_HOST}$1 [R=301,L]

</VirtualHost>
```

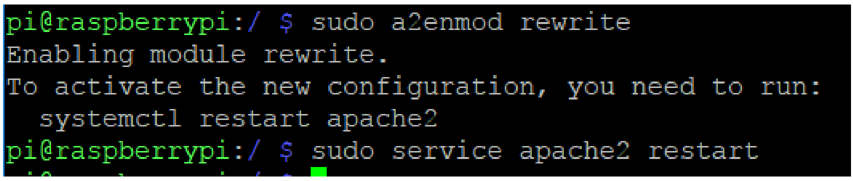
Image 23. New config file enforcing SSL connection

Redirect module

# sudo a2enmod rewrite

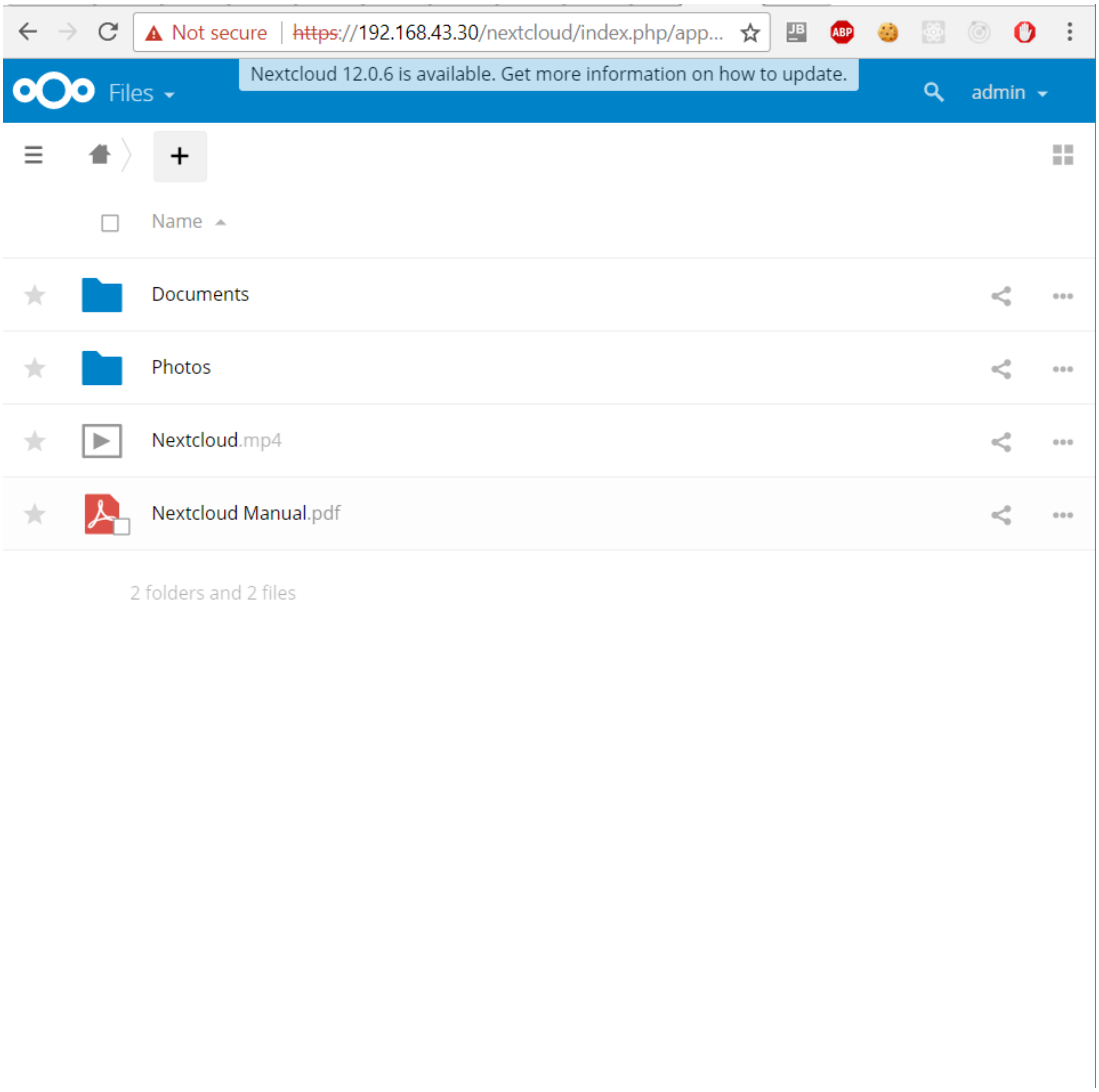
Restart Apache

# sudo service apache2 restart



```
pi@raspberrypi:/ $ sudo a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
  systemctl restart apache2
pi@raspberrypi:/ $ sudo service apache2 restart
```

Image 24. Redirect module and restart Apache2



Not secure | ~~https~~://192.168.43.30/nextcloud/index.php/app...

Nextcloud 12.0.6 is available. Get more information on how to update.

Documents

Photos

Nextcloud.mp4

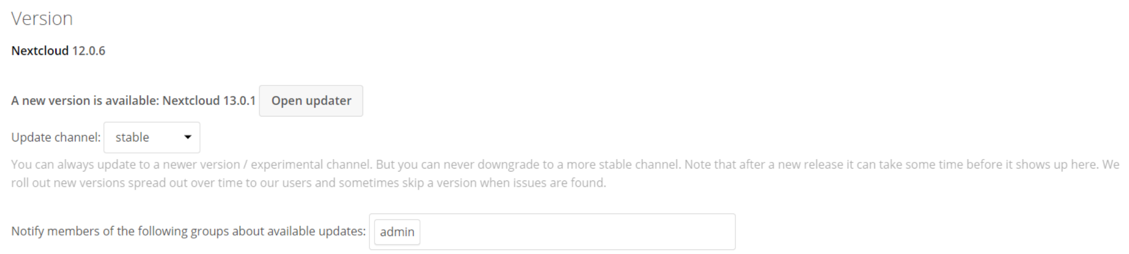
Nextcloud Manual.pdf

2 folders and 2 files

Image 25. Result – all traffic is using https instead of http

Appendix 5: Upgrade NextCloud

Step 1. Go to admin and open updater



Version

Nextcloud 12.0.6

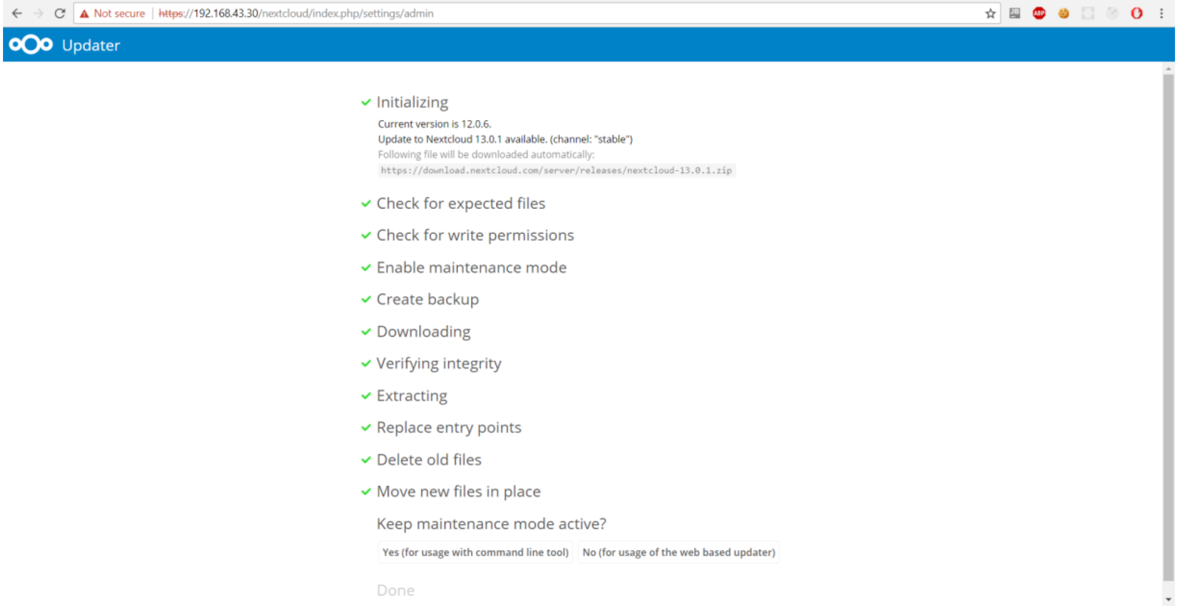
A new version is available: Nextcloud 13.0.1 Open updater

Update channel: stable

You can always update to a newer version / experimental channel. But you can never downgrade to a more stable channel. Note that after a new release it can take some time before it shows up here. We roll out new versions spread out over time to our users and sometimes skip a version when issues are found.

Notify members of the following groups about available updates: admin

Image 26. Open updater in admin session



Not secure
~~https~~://192.168.43.30/nextcloud/index.php/settings/admin
Updater

✓ Initializing

Current version is 12.0.6.
Update to Nextcloud 13.0.1 available. (channel: "stable")
Following file will be downloaded automatically:
`https://download.nextcloud.com/server/releases/nextcloud-13.0.1.zip`

✓ Check for expected files

✓ Check for write permissions

✓ Enable maintenance mode

✓ Create backup

✓ Downloading

✓ Verifying integrity

✓ Extracting

✓ Replace entry points

✓ Delete old files

✓ Move new files in place

Keep maintenance mode active?

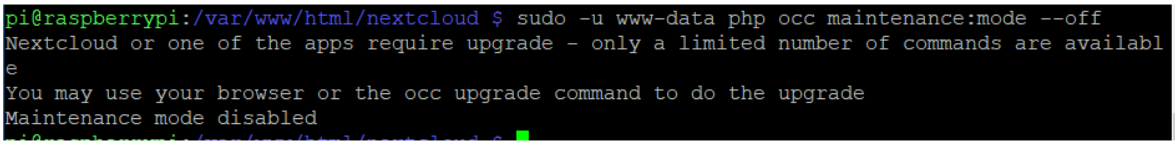
Yes (for usage with command line tool) No (for usage of the web based updater)

Done

Image 27. Updating from version 11.0.2 to version 12.0.6

Step 2. Turn off maintenance mode

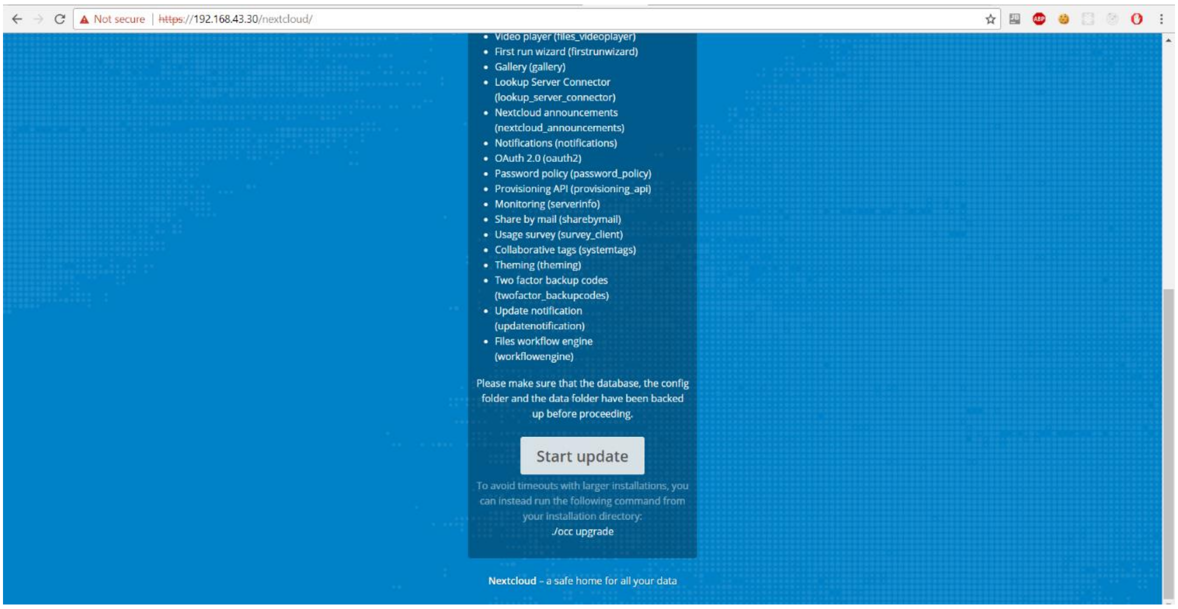
# sudo -u www-data php occ maintenance:mode --off



```
pi@raspberrypi:/var/www/html/nextcloud $ sudo -u www-data php occ maintenance:mode --off
Nextcloud or one of the apps require upgrade - only a limited number of commands are availabl
e
You may use your browser or the occ upgrade command to do the upgrade
Maintenance mode disabled
pi@raspberrypi:/var/www/html/nextcloud $
```

Image 28. Turn off maintenance mode

Step 3. Start new update



Not secure | https://192.168.43.30/nextcloud/

- Video player (files_videoplayer)
- First run wizard (firstrunwizard)
- Gallery (gallery)
- Lookup Server Connector (lookup_server_connector)
- Nextcloud announcements (nextcloud_announcements)
- Notifications (notifications)
- OAuth 2.0 (oauth2)
- Password policy (password_policy)
- Provisioning API (provisioning_api)
- Monitoring (serverinfo)
- Share by mail (sharebymail)
- Usage survey (survey_client)
- Collaborative tags (systemtags)
- Theming (theming)
- Two factor backup codes (twofactor_backupcodes)
- Update notification (updatenotification)
- Files workflow engine (workflowengine)

Please make sure that the database, the config folder and the data folder have been backed up before proceeding.

To avoid timeouts with larger installations, you can instead run the following command from your installation directory:

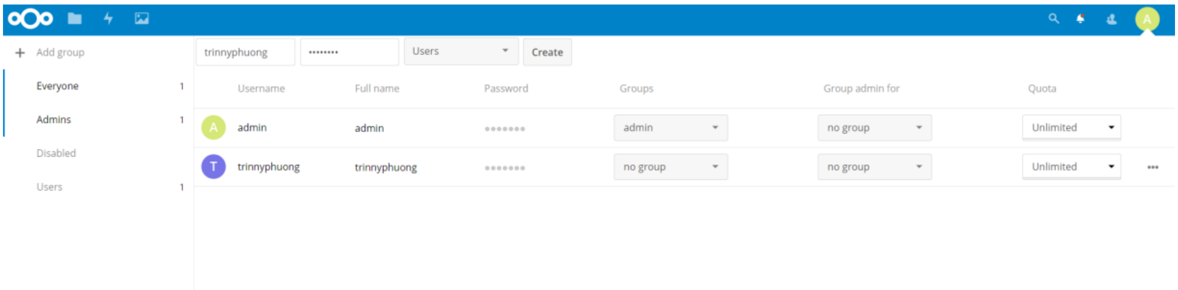
```
./occ upgrade
```

Nextcloud – a safe home for all your data

Image 29. Start new update

Appendix 6: NextCloud client installation

Step 1. Create a new user in the server-side



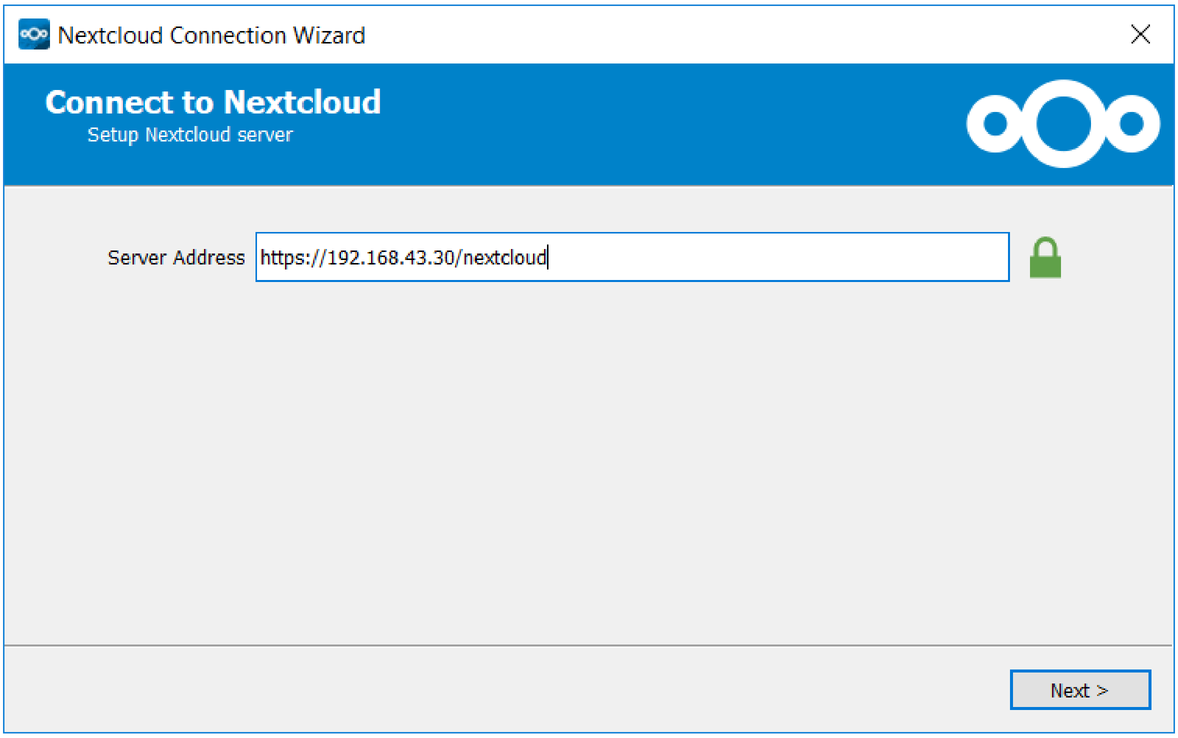
trinnyphuong | •••••••• | Users

Group	
Everyone	1
Admins	1
Disabled	
Users	1

	Username	Full name	Password	Groups	Group admin for	Quota
A	admin	admin	•••••••	admin	no group	Unlimited
T	trinnyphuong	trinnyphuong	•••••••	no group	no group	Unlimited

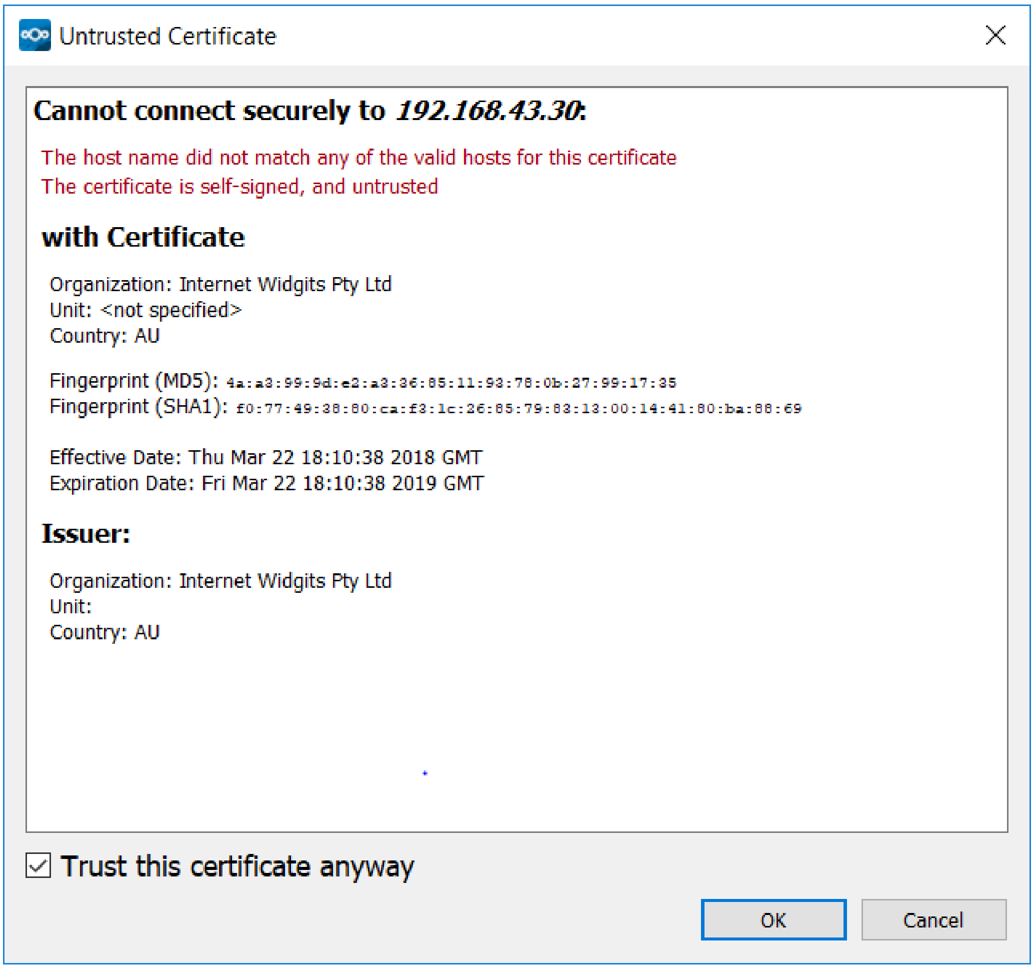
Image 30. Creating a new user in the server-side

Step 2. Download the client from NextCloud



Server Address: https://192.168.43.30/nextcloud

Image 31. Entering the domain address of the server



Cannot connect securely to *192.168.43.30*:

The host name did not match any of the valid hosts for this certificate
The certificate is self-signed, and untrusted

with Certificate

Organization: Internet Widgits Pty Ltd
Unit: <not specified>
Country: AU

Fingerprint (MD5): 4a:a3:99:9d:e2:a3:36:85:11:93:78:0b:27:99:17:35
Fingerprint (SHA1): f0:77:49:38:80:ca:f3:1c:26:85:79:83:13:00:14:41:80:ba:88:69

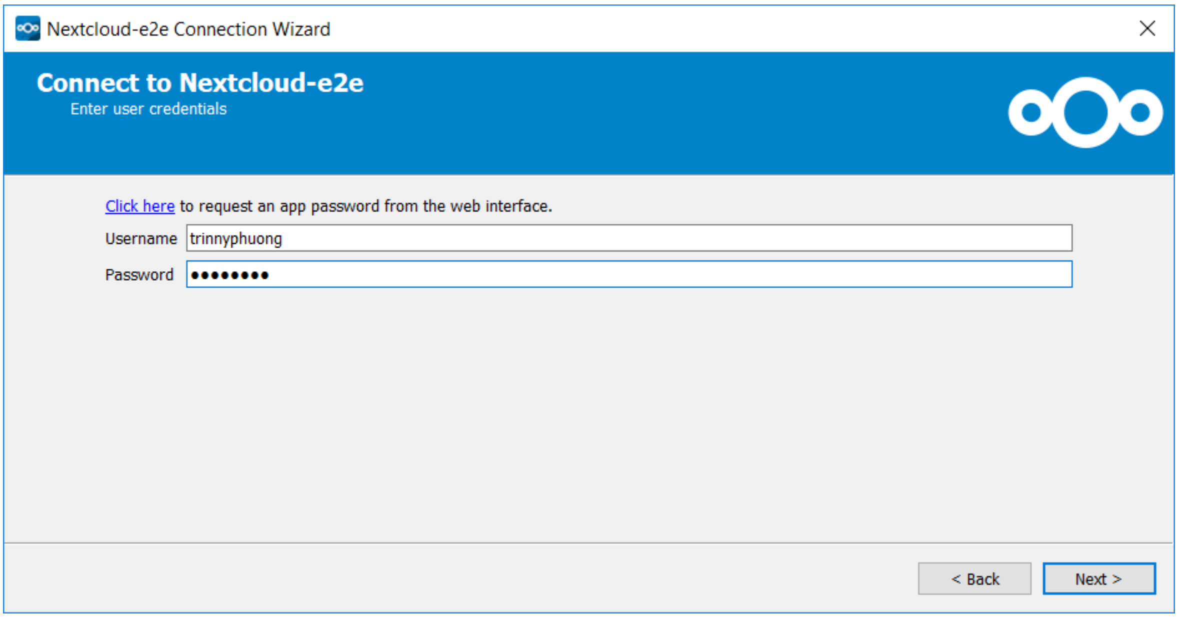
Effective Date: Thu Mar 22 18:10:38 2018 GMT
Expiration Date: Fri Mar 22 18:10:38 2019 GMT

Issuer:

Organization: Internet Widgits Pty Ltd
Unit:
Country: AU

☑ Trust this certificate anyway

Image 32. Trust self-signed certificate


Nextcloud-e2e Connection Wizard

Connect to Nextcloud-e2e

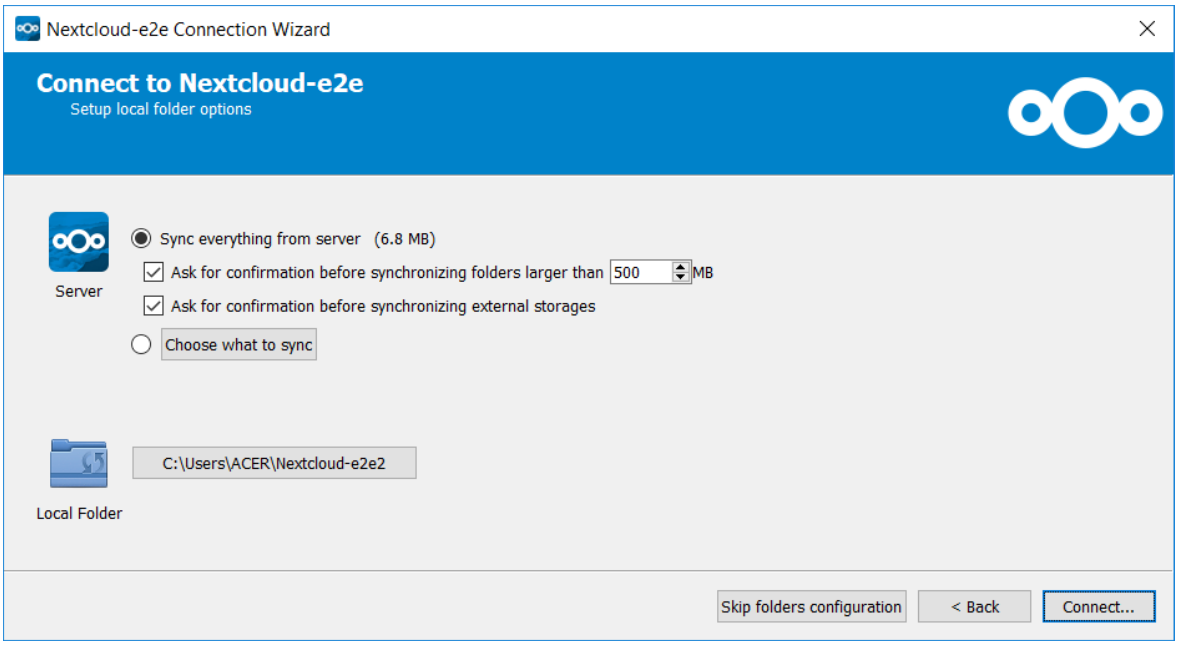
Enter user credentials

Click here to request an app password from the web interface.

Username: trinnyphuong

Password: ●●●●●●●●

Image 33. Enter user’s credentials


Nextcloud-e2e Connection Wizard

Connect to Nextcloud-e2e

Setup local folder options

Server

● Sync everything from server (6.8 MB)

☑ Ask for confirmation before synchronizing folders larger than 500 MB

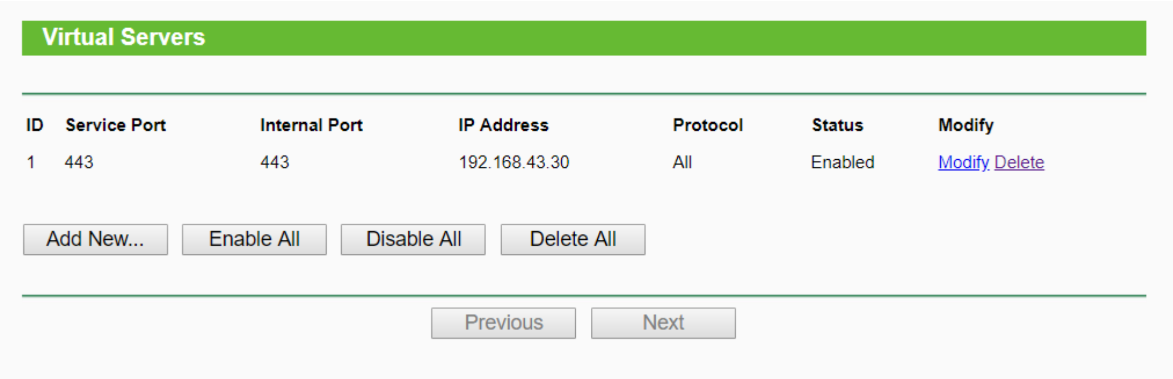
☑ Ask for confirmation before synchronizing external storages

○ Choose what to sync

Local Folder: C:\Users\ACER\Nextcloud-e2e2

Image 34. Choose sync options and connect

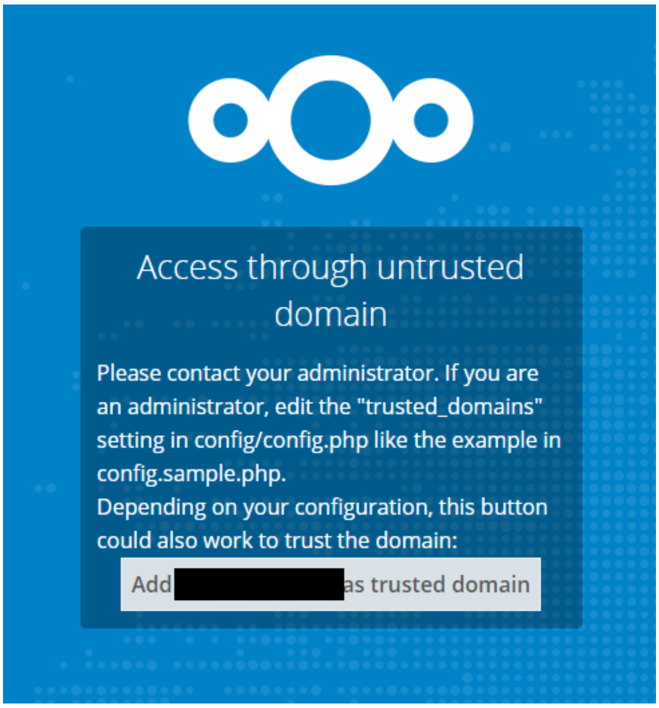
Appendix 7. Port forwarding



Virtual Servers

ID	Service Port	Internal Port	IP Address	Protocol	Status	Modify
1	443	443	192.168.43.30	All	Enabled	Modify Delete

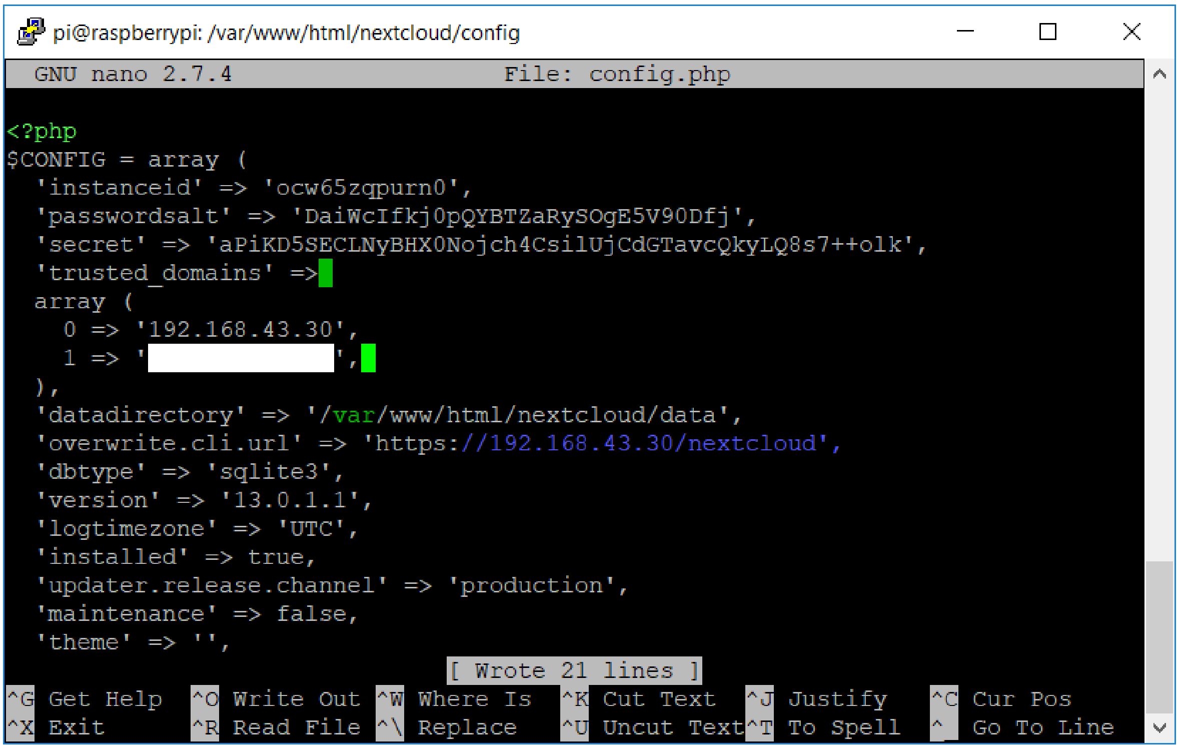
Image 35. Port forwarding on router



Access through untrusted domain

Please contact your administrator. If you are an administrator, edit the "trusted_domains" setting in config/config.php like the example in config.sample.php.
Depending on your configuration, this button could also work to trust the domain:

Image 36. Domain needs to be trusted warning



```
pi@raspberrypi: /var/www/html/nextcloud/config

  GNU nano 2.7.4                    File: config.php

<?php
$CONFIG = array (
  'instanceid' => 'ocw65zqpurn0',
  'passwordsalt' => 'DaiWcIfkj0pQYBTZaRySOgE5V90Dfj',
  'secret' => 'aPiKD5SECLNyBHX0Nojch4CsilUjCdGTavcQkyLQ8s7++olk',
  'trusted_domains' =>
  array (
    0 => '192.168.43.30',
    1 => '[redacted]',
  ),
  'datadirectory' => '/var/www/html/nextcloud/data',
  'overwrite.cli.url' => 'https://192.168.43.30/nextcloud',
  'dbtype' => 'sqlite3',
  'version' => '13.0.1.1',
  'logtimezone' => 'UTC',
  'installed' => true,
  'updater.release.channel' => 'production',
  'maintenance' => false,
  'theme' => '',
                               [ Wrote 21 lines ]
^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^\ Replace    ^U Uncut Text^T To Spell    ^_ Go To Line
```

Image 37. Adding trusted domain manually in config.php.

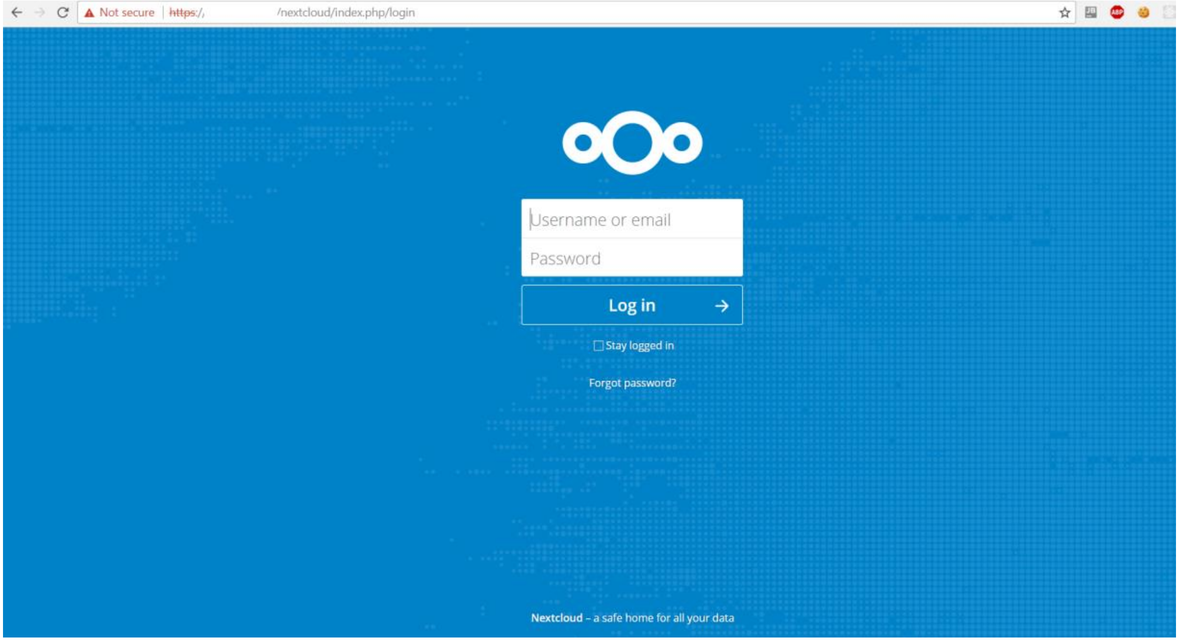
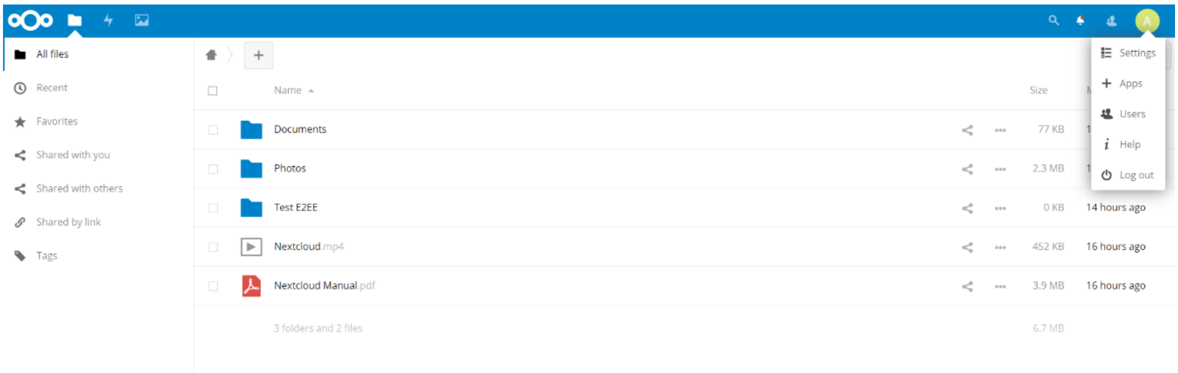

Not secure
https://   /nextcloud/index.php/login
Username or email
Password
Log in
Stay logged in
Forgot password?
Nextcloud – a safe home for all your data

Image 38. External login.

Appendix 8: User test

Step 1. Go to apps



☐	Name ▲			Size	
☐	Documents	<	•••	77 KB	1
☐	Photos	<	•••	2.3 MB	1
☐	Test E2EE	<	•••	0 KB	14 hours ago
☐	Nextcloud.mp4	<	•••	452 KB	16 hours ago
☐	Nextcloud Manual.pdf	<	•••	3.9 MB	16 hours ago
	3 folders and 2 files			6.7 MB	

Image 39. NextCloud's UI

Step 2. Enable Server-side encryption (Default Encryption Module) and End-to-End Encryption

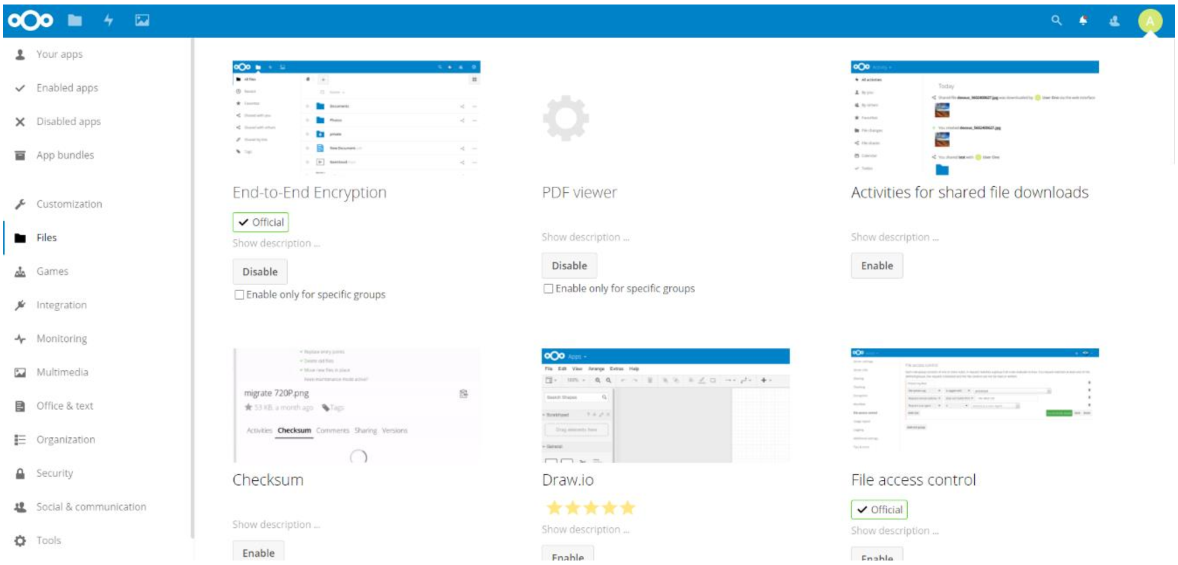
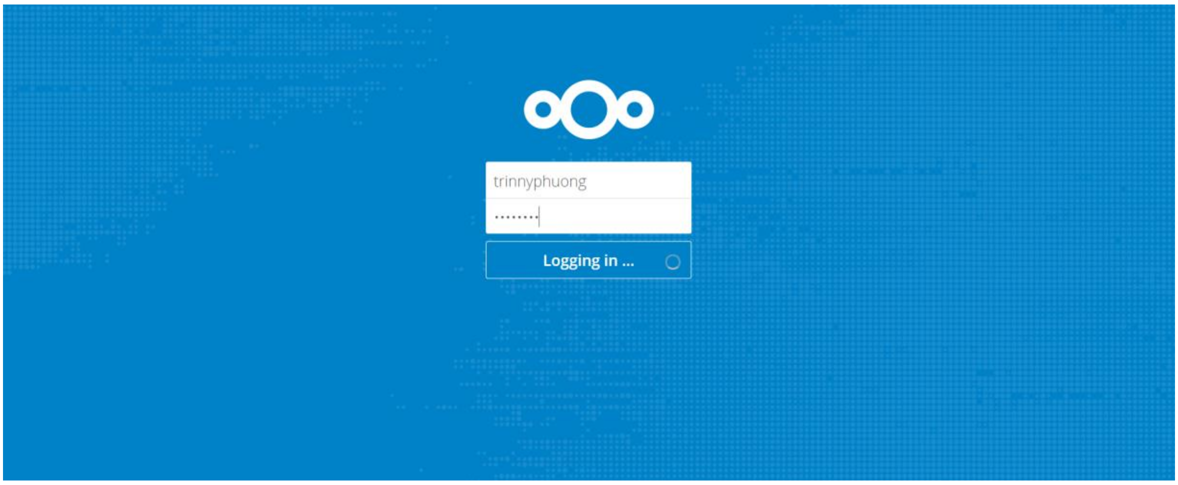
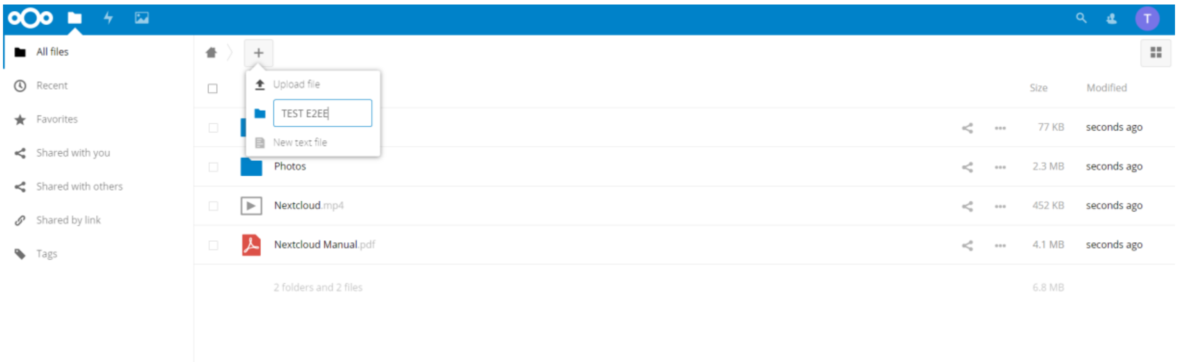


Image 40. Enable End-to-End Encryption


trinnyphuong

••••••••

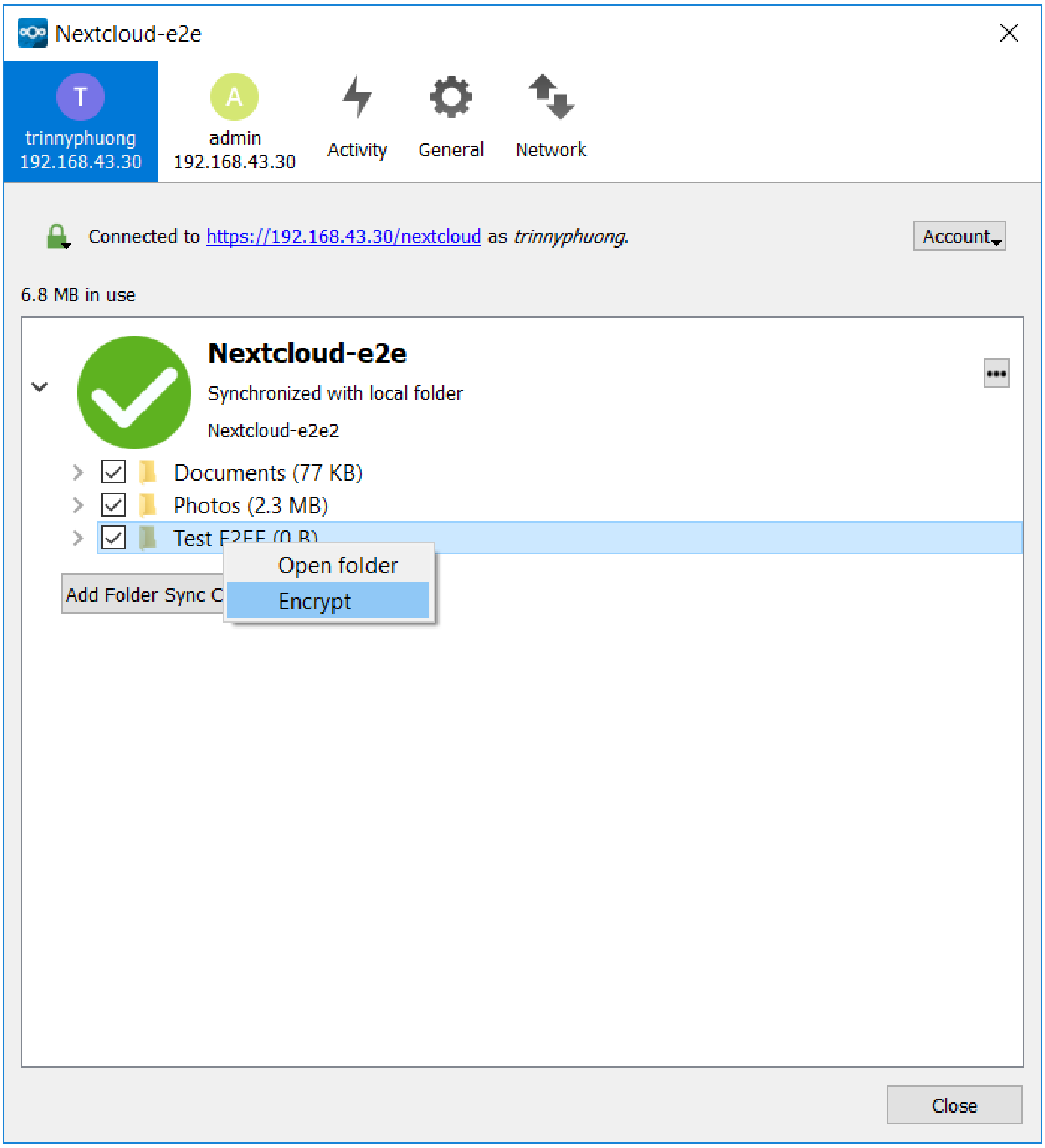
Image 41. Log in as a user



TEST E2EE

Name			Size	Modified
		•••	77 KB	seconds ago
Photos		•••	2.3 MB	seconds ago
Nextcloud.mp4		•••	452 KB	seconds ago
Nextcloud Manual.pdf		•••	4.1 MB	seconds ago
2 folders and 2 files			6.8 MB	

Image 42. Create test folder



Connected to https://192.168.43.30/nextcloud as *trinnyphuong*.

6.8 MB in use

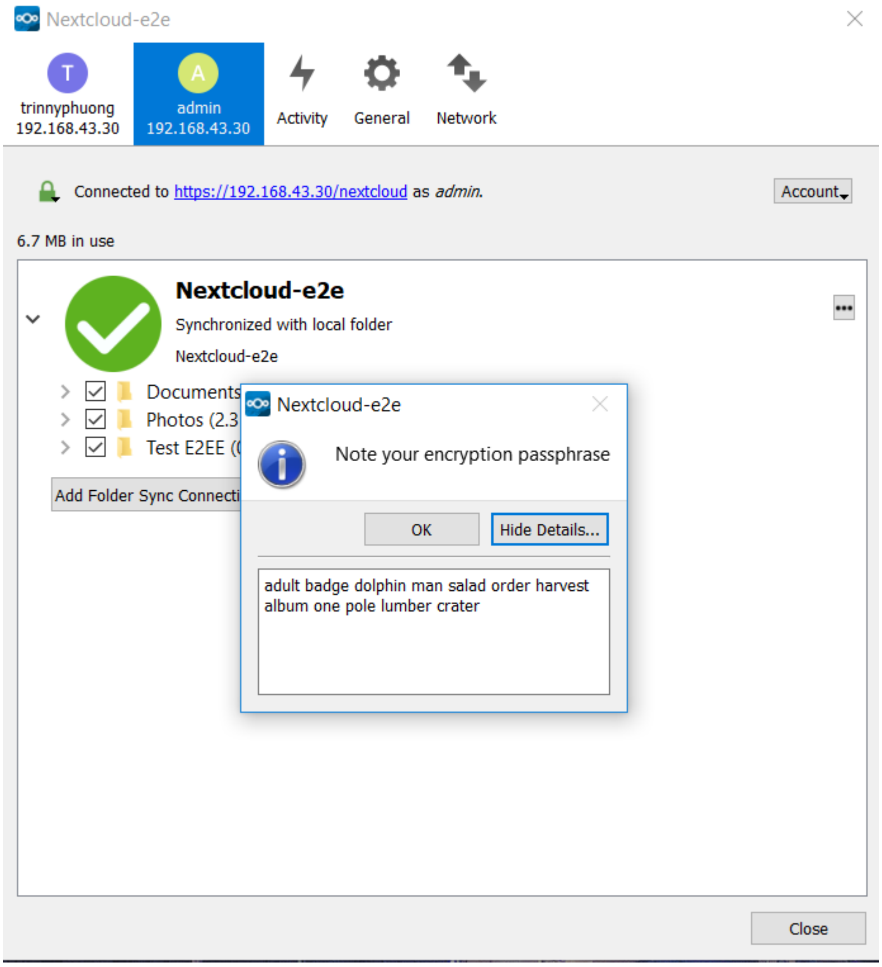
Nextcloud-e2e

Synchronized with local folder

Nextcloud-e2e2

- ☑ Documents (77 KB)
- ☑ Photos (2.3 MB)
- ☑ Test E2EE (0 B)

Image 43. Encrypt the folder on the client-side



Connected to https://192.168.43.30/nextcloud as *admin*.

6.7 MB in use

Nextcloud-e2e
Synchronized with local folder
Nextcloud-e2e

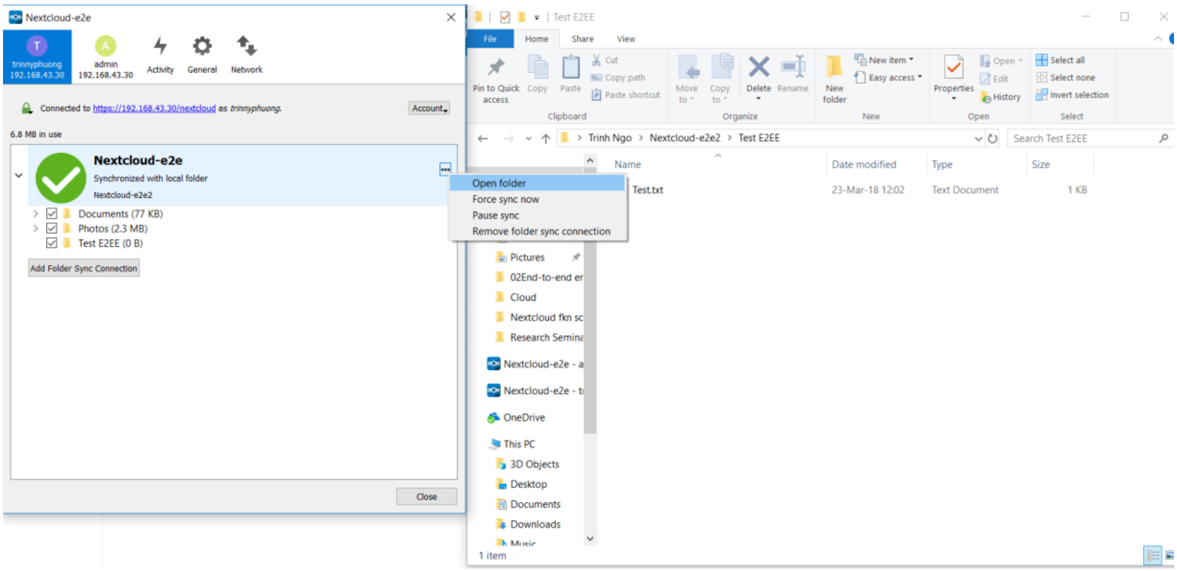
Nextcloud-e2e

Note your encryption passphrase

adult badge dolphin man salad order harvest album one pole lumber crater

Image 44. The 12-word long mnemonic

The folder is accessible over the client and it contains a Test.txt file



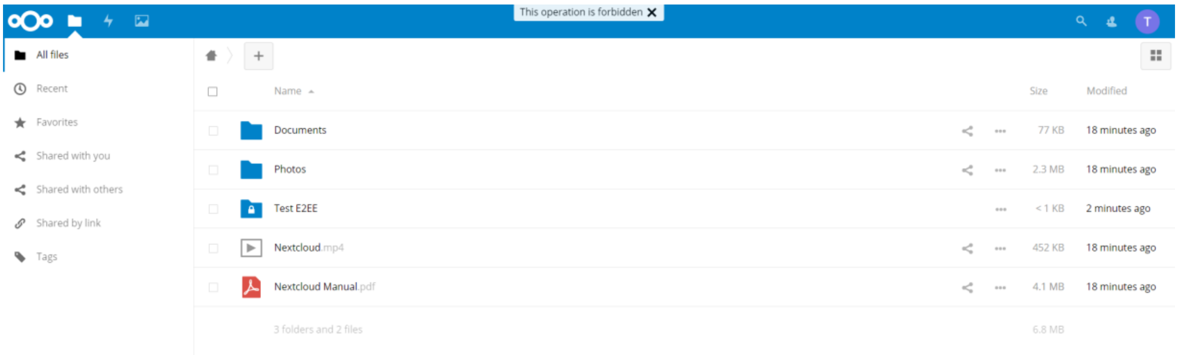
Connected to https://192.168.43.30/nextcloud as *trinnyphuong*.

6.8 MB in use

Nextcloud-e2e
Synchronized with local folder
Nextcloud-e2e2

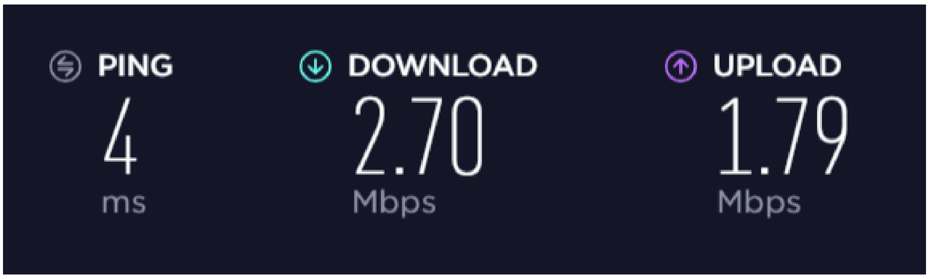
Name	Date modified	Type	Size
Test.txt	23-Mar-18 12:02	Text Document	1 KB

Image 45. Adding a text document in to the end-to-end encrypted folder.



	Name		Size	Modified
□	Documents		77 KB	18 minutes ago
□	Photos		2.3 MB	18 minutes ago
□	Test E2EE		< 1 KB	2 minutes ago
□	Nextcloud.mp4		452 KB	18 minutes ago
□	Nextcloud Manual.pdf		4.1 MB	18 minutes ago
	3 folders and 2 files		6.8 MB	

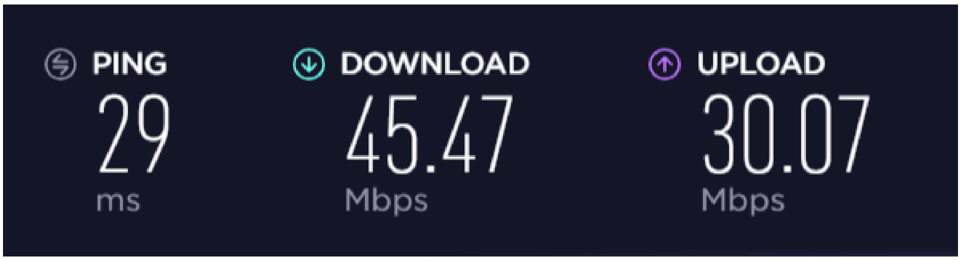
Image 46. End-to-end encrypted showed on the server-side


PING 4 ms

DOWNLOAD 2.70 Mbps

UPLOAD 1.79 Mbps

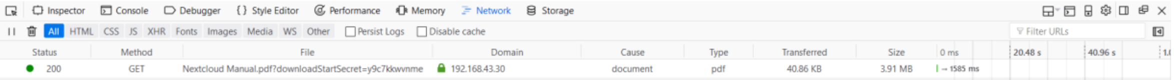
Image 47. Speed test inside the local network


PING 29 ms

DOWNLOAD 45.47 Mbps

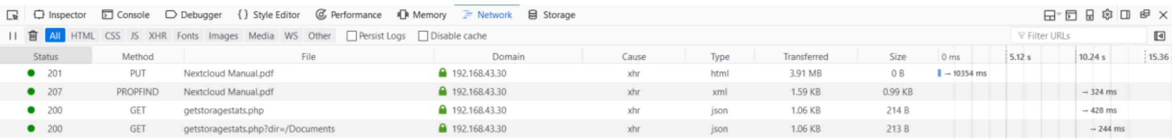
UPLOAD 30.07 Mbps

Image 48. Speed test outside the local network



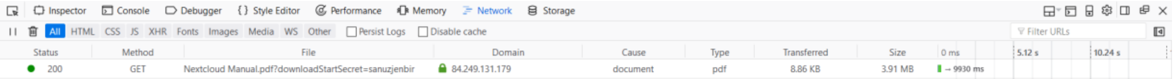
Status	Method	File	Domain	Cause	Type	Transferred	Size	0 ms
● 200	GET	Nextcloud Manual.pdf?downloadStartSecret=y9c7kkwvnme	192.168.43.30	document	pdf	40.86 KB	3.91 MB	→ 1585 ms

Image 49. Download PDF file (3.91MB) inside the local network



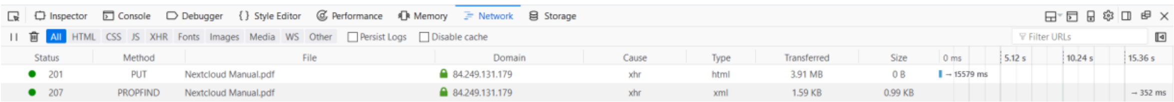
Status	Method	File	Domain	Cause	Type	Transferred	Size	0 ms
● 201	PUT	Nextcloud Manual.pdf	192.168.43.30	xhr	html	3.91 MB	0 B	→ 10354 ms
● 207	PROPFIND	Nextcloud Manual.pdf	192.168.43.30	xhr	xml	1.59 KB	0.99 KB	→ 324 ms
● 200	GET	getstoragestats.php	192.168.43.30	xhr	json	1.06 KB	214 B	→ 428 ms
● 200	GET	getstoragestats.php?dir=/Documents	192.168.43.30	xhr	json	1.06 KB	213 B	→ 244 ms

Image 50. Upload PDF file (3.91MB) inside the local network



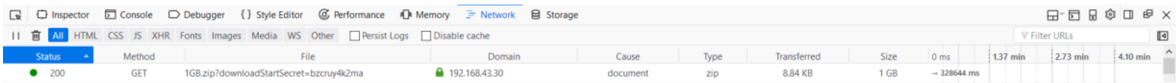
Status	Method	File	Domain	Cause	Type	Transferred	Size	0 ms
● 200	GET	Nextcloud Manual.pdf?downloadStartSecret=sanuzjenbir	84.249.131.179	document	pdf	8.86 KB	3.91 MB	→ 9930 ms

Image 51. Download PDF file (3.91MB) outside the local network



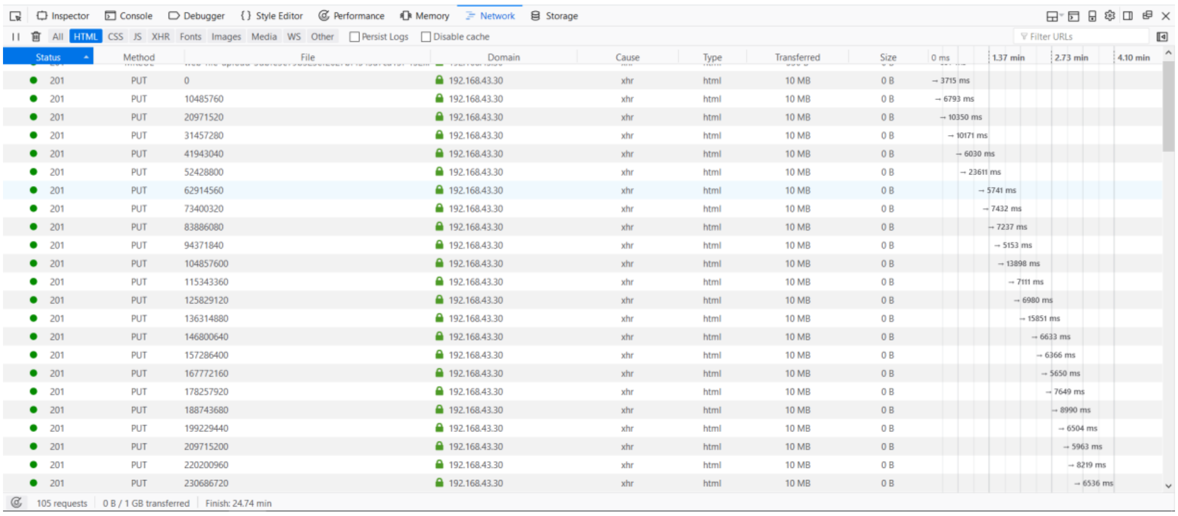
Status	Method	File	Domain	Cause	Type	Transferred	Size	0 ms 5.12 s 10.24 s 15.36 s
● 201	PUT	Nextcloud Manual.pdf	84.249.131.179	xhr	html	3.91 MB	0 B	→ 15579 ms
● 207	PROPFIND	Nextcloud Manual.pdf	84.249.131.179	xhr	xml	1.59 KB	0.99 KB	→ 352 ms

Image 52. Upload PDF file (3.91MB) outside the local network.



Status	Method	File	Domain	Cause	Type	Transferred	Size	0 ms 1.37 min 2.73 min 4.10 min
● 200	GET	1GB.zip?downloadStartSecret=bzcruy4k2ma	192.168.43.30	document	zip	8.84 KB	1 GB	→ 328644 ms

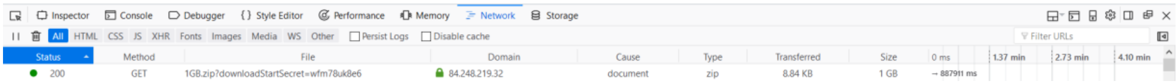
Image 53. Download ZIP file (1GB) inside the local network.



Status	Method	File	Domain	Cause	Type	Transferred	Size	0 ms 1.37 min 2.73 min 4.10 min
● 201	PUT	0	192.168.43.30	xhr	html	10 MB	0 B	→ 3715 ms
● 201	PUT	10485760	192.168.43.30	xhr	html	10 MB	0 B	→ 6793 ms
● 201	PUT	20971520	192.168.43.30	xhr	html	10 MB	0 B	→ 10350 ms
● 201	PUT	31457280	192.168.43.30	xhr	html	10 MB	0 B	→ 10171 ms
● 201	PUT	41943040	192.168.43.30	xhr	html	10 MB	0 B	→ 6030 ms
● 201	PUT	52428800	192.168.43.30	xhr	html	10 MB	0 B	→ 23611 ms
● 201	PUT	62914560	192.168.43.30	xhr	html	10 MB	0 B	→ 5741 ms
● 201	PUT	73400320	192.168.43.30	xhr	html	10 MB	0 B	→ 7432 ms
● 201	PUT	83886080	192.168.43.30	xhr	html	10 MB	0 B	→ 7237 ms
● 201	PUT	94371840	192.168.43.30	xhr	html	10 MB	0 B	→ 5153 ms
● 201	PUT	104857600	192.168.43.30	xhr	html	10 MB	0 B	→ 13898 ms
● 201	PUT	115343360	192.168.43.30	xhr	html	10 MB	0 B	→ 7111 ms
● 201	PUT	125829120	192.168.43.30	xhr	html	10 MB	0 B	→ 6980 ms
● 201	PUT	136314880	192.168.43.30	xhr	html	10 MB	0 B	→ 15851 ms
● 201	PUT	146800640	192.168.43.30	xhr	html	10 MB	0 B	→ 6633 ms
● 201	PUT	157286400	192.168.43.30	xhr	html	10 MB	0 B	→ 6366 ms
● 201	PUT	167772160	192.168.43.30	xhr	html	10 MB	0 B	→ 5650 ms
● 201	PUT	178257920	192.168.43.30	xhr	html	10 MB	0 B	→ 7649 ms
● 201	PUT	188743680	192.168.43.30	xhr	html	10 MB	0 B	→ 8990 ms
● 201	PUT	199229440	192.168.43.30	xhr	html	10 MB	0 B	→ 6504 ms
● 201	PUT	209715200	192.168.43.30	xhr	html	10 MB	0 B	→ 5963 ms
● 201	PUT	220200960	192.168.43.30	xhr	html	10 MB	0 B	→ 8219 ms
● 201	PUT	230686720	192.168.43.30	xhr	html	10 MB	0 B	→ 6536 ms

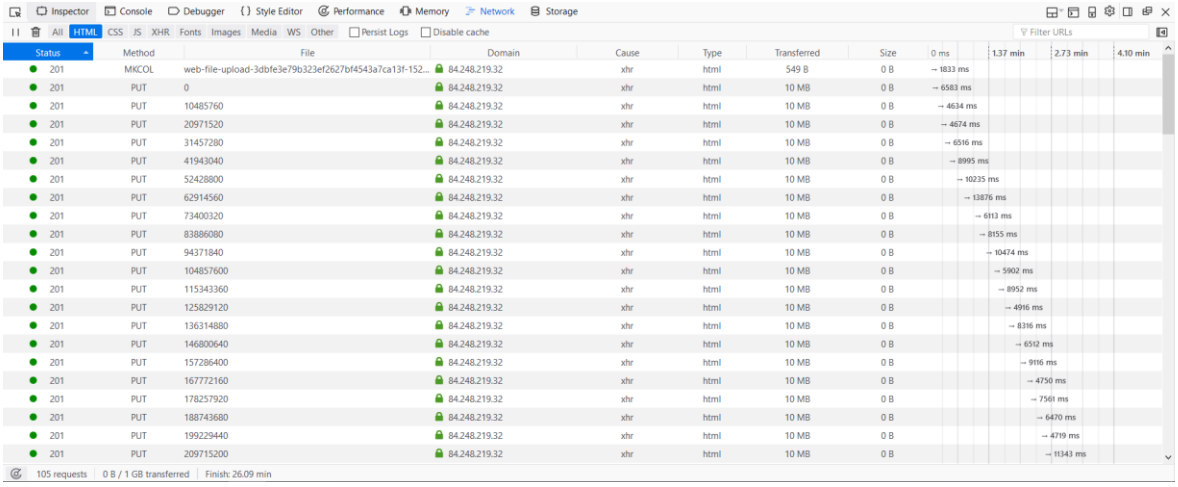
105 requests | 0 B / 1 GB transferred | Finish: 24.74 min

Image 54. Upload ZIP file (1GB) inside the local network.



Status	Method	File	Domain	Cause	Type	Transferred	Size	0 ms 1.37 min 2.73 min 4.10 min
● 200	GET	1GB.zip?downloadStartSecret=wfm78uk8e6	84.248.219.32	document	zip	8.84 KB	1 GB	→ 887911 ms

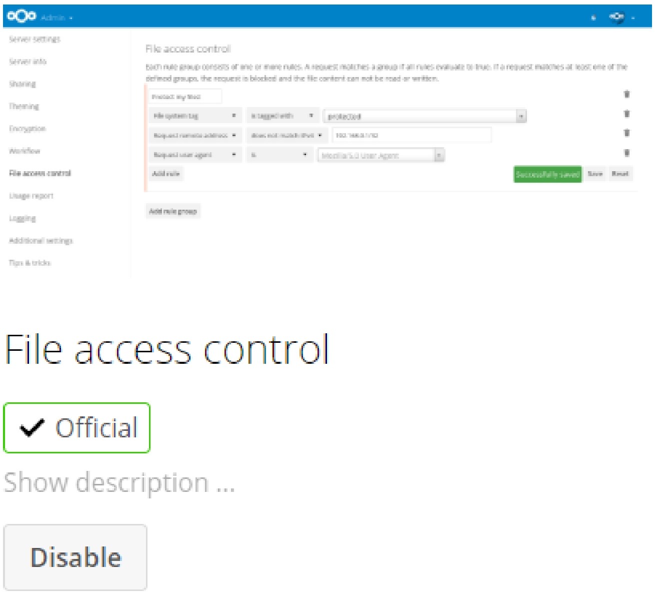
Image 55. Download ZIP file (1GB) outside the local network.



Status	Method	File	Domain	Cause	Type	Transferred	Size	0 ms 1.37 min 2.73 min 4.10 min
● 201	MKCOL	web-file-upload-3dbfe3e79b323ef2627bf4543a7ca13f-152...	84.248.219.32	xhr	html	549 B	0 B	→ 1833 ms
● 201	PUT	0	84.248.219.32	xhr	html	10 MB	0 B	→ 6583 ms
● 201	PUT	10485760	84.248.219.32	xhr	html	10 MB	0 B	→ 4634 ms
● 201	PUT	20971520	84.248.219.32	xhr	html	10 MB	0 B	→ 4674 ms
● 201	PUT	31457280	84.248.219.32	xhr	html	10 MB	0 B	→ 6516 ms
● 201	PUT	41943040	84.248.219.32	xhr	html	10 MB	0 B	→ 8995 ms
● 201	PUT	52428800	84.248.219.32	xhr	html	10 MB	0 B	→ 10235 ms
● 201	PUT	62914560	84.248.219.32	xhr	html	10 MB	0 B	→ 13876 ms
● 201	PUT	73400320	84.248.219.32	xhr	html	10 MB	0 B	→ 6113 ms
● 201	PUT	83886080	84.248.219.32	xhr	html	10 MB	0 B	→ 8155 ms
● 201	PUT	94371840	84.248.219.32	xhr	html	10 MB	0 B	→ 10474 ms
● 201	PUT	104857600	84.248.219.32	xhr	html	10 MB	0 B	→ 5902 ms
● 201	PUT	115343360	84.248.219.32	xhr	html	10 MB	0 B	→ 8952 ms
● 201	PUT	125829120	84.248.219.32	xhr	html	10 MB	0 B	→ 4916 ms
● 201	PUT	136314880	84.248.219.32	xhr	html	10 MB	0 B	→ 8316 ms
● 201	PUT	146800640	84.248.219.32	xhr	html	10 MB	0 B	→ 6512 ms
● 201	PUT	157286400	84.248.219.32	xhr	html	10 MB	0 B	→ 9116 ms
● 201	PUT	167772160	84.248.219.32	xhr	html	10 MB	0 B	→ 4750 ms
● 201	PUT	178257920	84.248.219.32	xhr	html	10 MB	0 B	→ 7561 ms
● 201	PUT	188743680	84.248.219.32	xhr	html	10 MB	0 B	→ 6470 ms
● 201	PUT	199229440	84.248.219.32	xhr	html	10 MB	0 B	→ 4719 ms
● 201	PUT	209715200	84.248.219.32	xhr	html	10 MB	0 B	→ 11343 ms

105 requests | 0 B / 1 GB transferred | Finish: 26.09 min

Image 56. Upload ZIP file (1GB) outside the local network.



Server settings
Server info
Sharing
Theming
Encryption
Workflow
File access control
Usage report
Logging
Additional settings
Tips & tricks

File access control

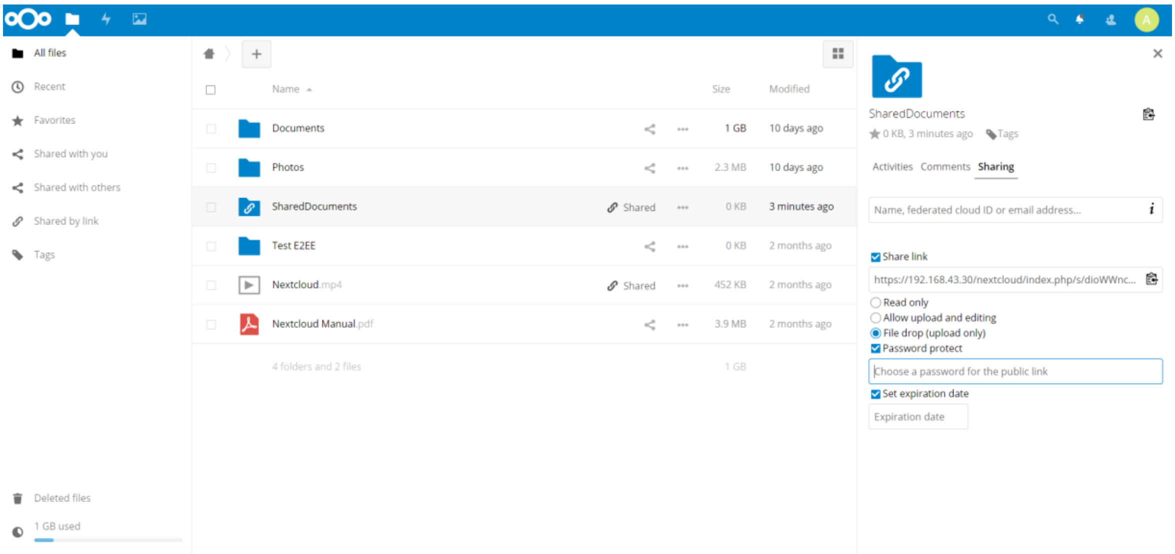
Each rule group consists of one or more rules. A request matches a group if all rules evaluate to true. If a request matches at least one of the defined groups, the request is blocked and the file content can not be read or written.

Protect my files

File system tag	is tagged with	protected
Request remote address	does not match IPv4	192.168.0.1/32
Request user agent	is	Mozilla/5.0 User Agent

File access control

Image 57. Enable File Access Control.



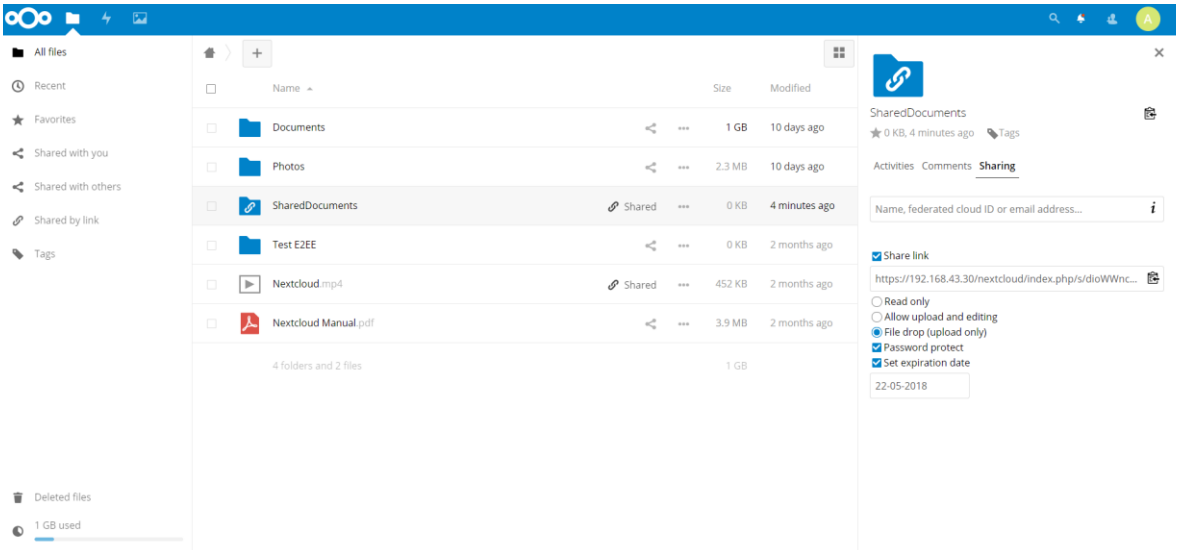
Name		Size	Modified
Documents		1 GB	10 days ago
Photos		2.3 MB	10 days ago
SharedDocuments	Shared	0 KB	3 minutes ago
Test E2EE		0 KB	2 months ago
Nextcloud.mp4	Shared	452 KB	2 months ago
Nextcloud Manual.pdf		3.9 MB	2 months ago
4 folders and 2 files		1 GB	

SharedDocuments
★ 0 KB, 3 minutes ago Tags

Name, federated cloud ID or email address...

☑ Share link
https://192.168.43.30/nextcloud/index.php/s/dioWWnc...
○ Read only
○ Allow upload and editing
● File drop (upload only)
☑ Password protect
Choose a password for the public link
☑ Set expiration date
Expiration date

Image 58. Sharing a folder.



	Name			Size	Modified
□	Documents		•••	1 GB	10 days ago
□	Photos		•••	2.3 MB	10 days ago
□	SharedDocuments	Shared	•••	0 KB	4 minutes ago
□	Test E2EE		•••	0 KB	2 months ago
□	Nextcloud.mp4	Shared	•••	452 KB	2 months ago
□	Nextcloud Manual.pdf		•••	3.9 MB	2 months ago
	4 folders and 2 files			1 GB	

SharedDocuments

0 KB, 4 minutes ago Tags

Activities Comments **Sharing**

Name, federated cloud ID or email address...

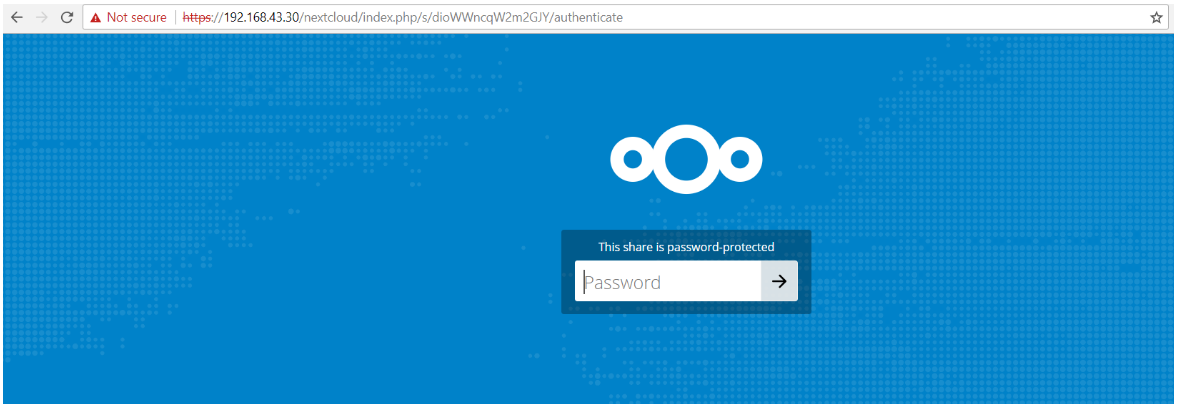
☑ Share link

https://192.168.43.30/nextcloud/index.php/s/dioWWnc...

○ Read only
○ Allow upload and editing
● File drop (upload only)
☑ Password protect
☑ Set expiration date

22-05-2018

Image 59. Sharing a folder (after entering password and expiration date).

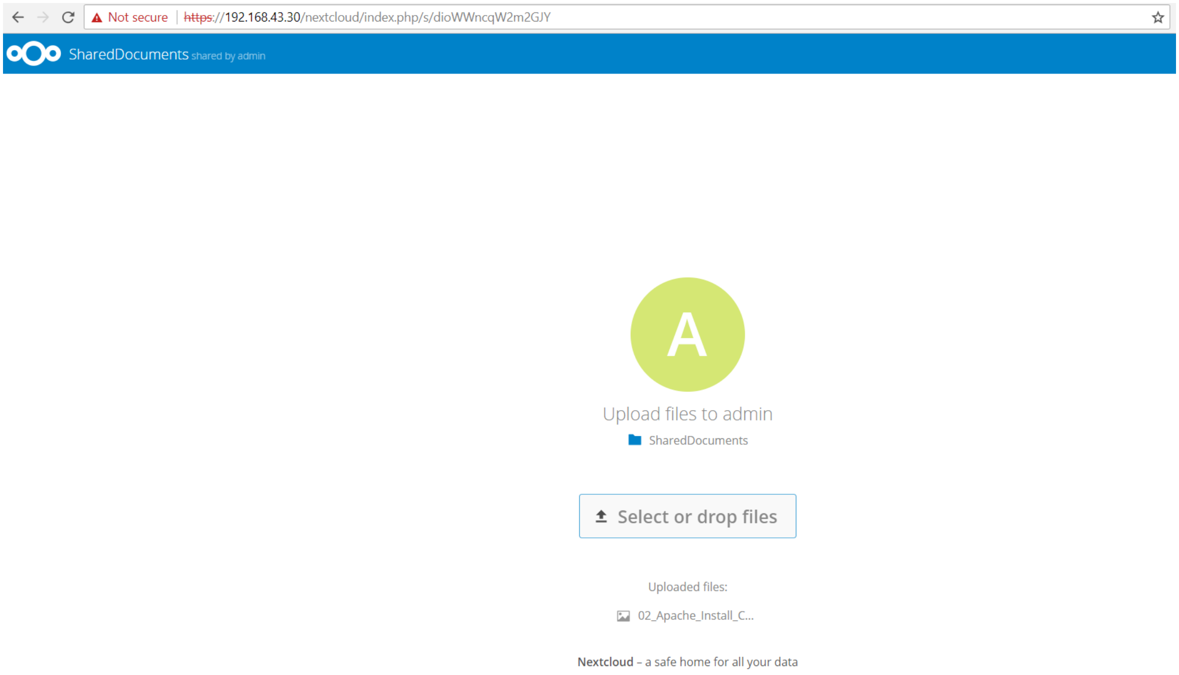


Not secure | ~~https~~://192.168.43.30/nextcloud/index.php/s/dioWWncqW2m2GJY/authenticate

This share is password-protected

Password

Image 60. Opening a shared folder



Not secure | ~~https~~://192.168.43.30/nextcloud/index.php/s/dioWWncqW2m2GJY

SharedDocuments shared by admin

Upload files to admin

SharedDocuments

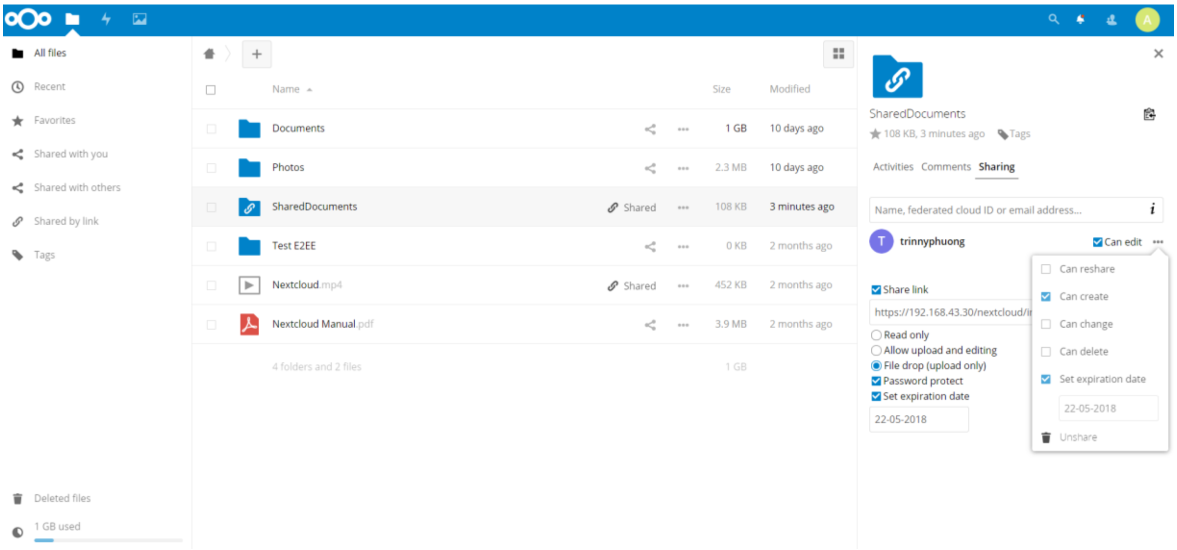
Select or drop files

Uploaded files:

02_Apache_Install_C...

Nextcloud – a safe home for all your data

Image 61. Secure file drop.



All files
Recent
Favorites
Shared with you
Shared with others
Shared by link
Tags
Deleted files
1 GB used

Name		Size	Modified
Documents		1 GB	10 days ago
Photos		2.3 MB	10 days ago
SharedDocuments	Shared	108 KB	3 minutes ago
Test E2EE		0 KB	2 months ago
Nextcloud.mp4	Shared	452 KB	2 months ago
Nextcloud Manual.pdf		3.9 MB	2 months ago
4 folders and 2 files		1 GB	

SharedDocuments
108 KB, 3 minutes ago Tags

Activities Comments **Sharing**

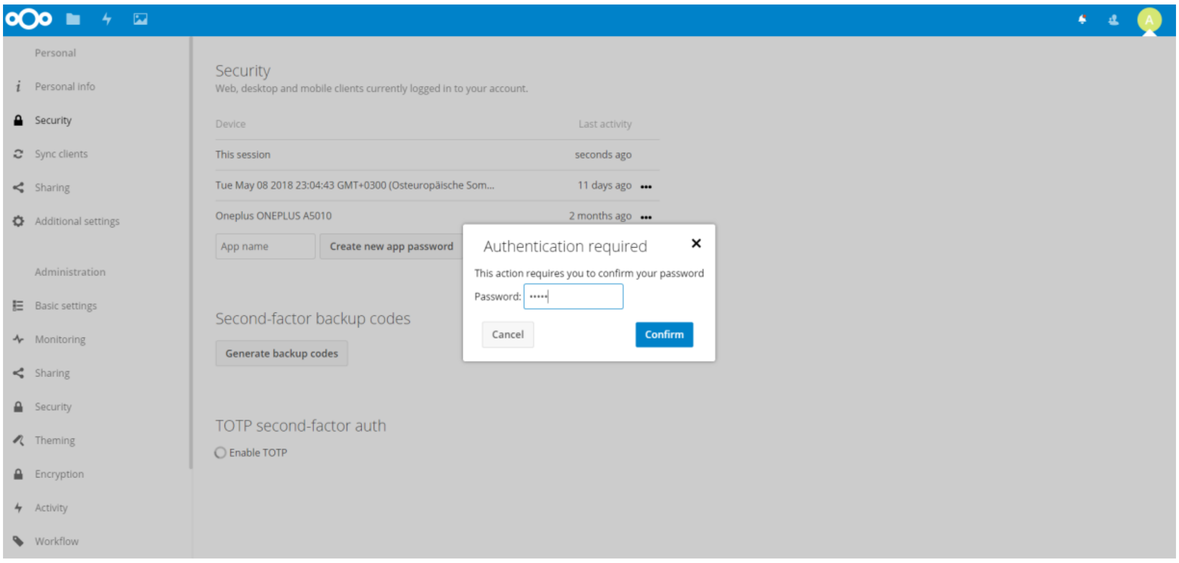
Name, federated cloud ID or email address...

trinnyphuong ☑ Can edit

- □ Can reshare
- ☑ Can create
- □ Can change
- □ Can delete
- ☑ Set expiration date
- 22-05-2018
- Unshare

☑ Share link
https://192.168.43.30/nextcloud/i
○ Read only
○ Allow upload and editing
● File drop (upload only)
☑ Password protect
☑ Set expiration date
22-05-2018

Image 62. Permission granted to shared user.



Personal
Personal info
Security
Sync clients
Sharing
Additional settings

Administration
Basic settings
Monitoring
Sharing
Security
Theming
Encryption
Activity
Workflow

Security
Web, desktop and mobile clients currently logged in to your account.

Device	Last activity
This session	seconds ago
Tue May 08 2018 23:04:43 GMT+0300 (Osteuropäische Som...	11 days ago
Oneplus ONEPLUS A5010	2 months ago

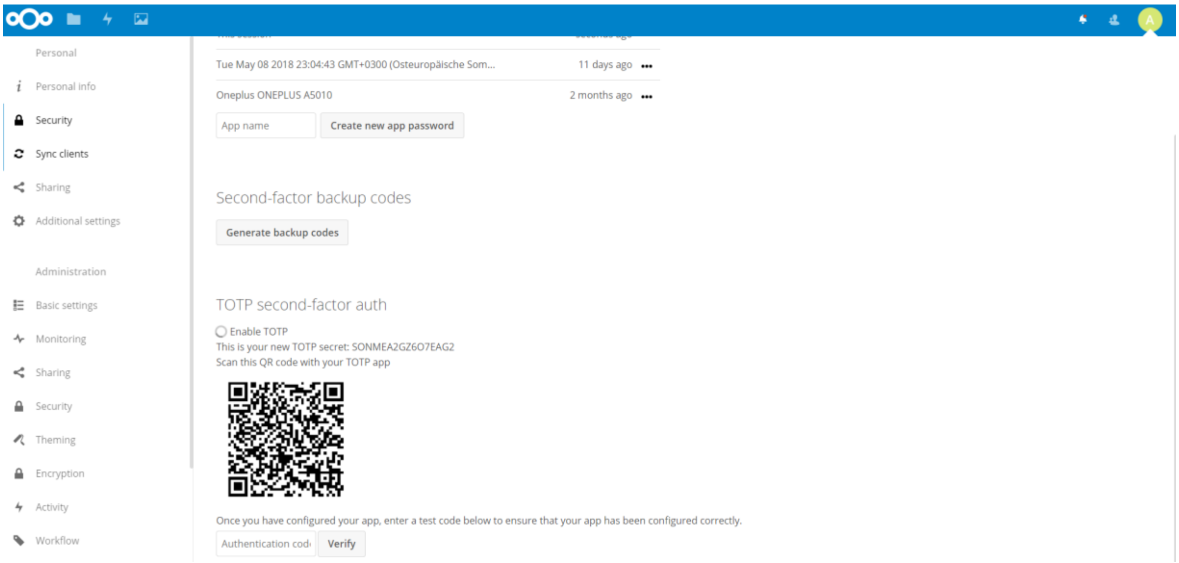
App name | Create new app password

Second-factor backup codes
Generate backup codes

TOTP second-factor auth
○ Enable TOTP

Authentication required
This action requires you to confirm your password
Password: •••••
Cancel | Confirm

Image 63. Set up back-up password.



Tue May 08 2018 23:04:43 GMT+0300 (Osteuropäische Som... 11 days ago •••

Oneplus ONEPLUS A5010 2 months ago •••

Second-factor backup codes

TOTP second-factor auth

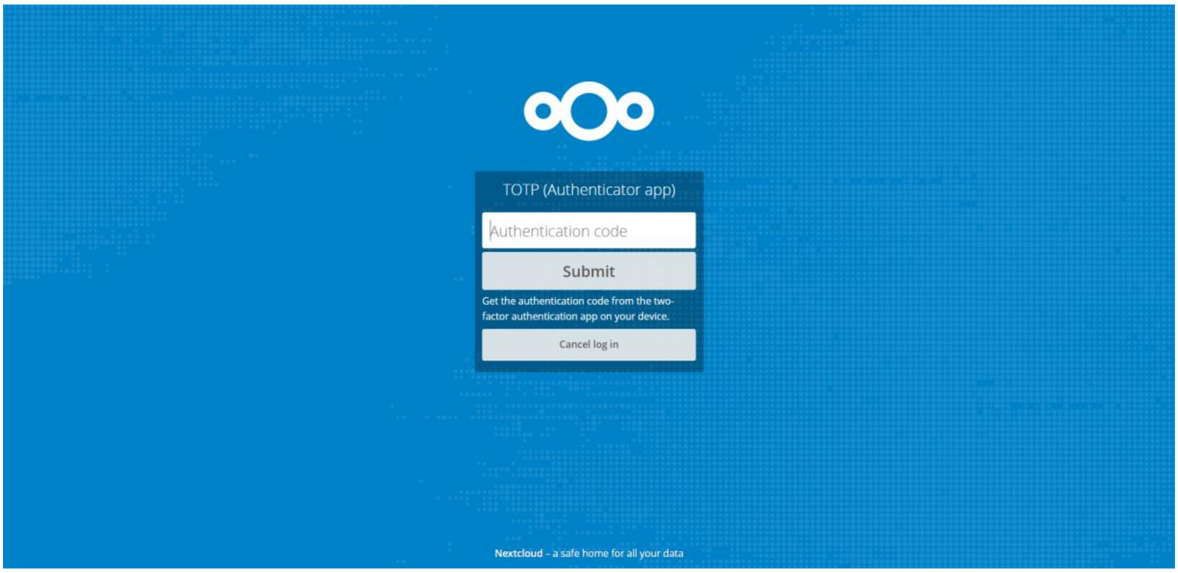
○ Enable TOTP

This is your new TOTP secret: SONMEA2GZ6O7EAG2

Scan this QR code with your TOTP app

Once you have configured your app, enter a test code below to ensure that your app has been configured correctly.

Image 64. Enable Two-factor Authentication Module (TOTP TwoFactor), QR code ready to be scanned.



TOTP (Authenticator app)

Get the authentication code from the two-factor authentication app on your device.

Nextcloud – a safe home for all your data

Image 65. Login after two-factor authentication has been set.