

**SIT TECHNICAL REPORTS**

# **ON THE SECURITY OF CLOUD STORAGE SERVICES**

03/2012





**Fraunhofer**  
SIT

# **On the Security of Cloud Storage Services**

Moritz Borgmann, Tobias Hahn, Michael Herfert, Thomas Kunz,  
Marcel Richter, Ursula Viebeg, Sven Vowé

Ed. Michael Waidner

SIT Technical Reports  
SIT-TR-2012-001

March 2012

Fraunhofer Institute for Secure  
Information Technology SIT  
Rheinstraße 75  
64295 Darmstadt  
Germany

# IMPRINT

## Contact

Fraunhofer Institute for  
Secure Information Technology SIT  
Rheinstraße 75  
64295 Darmstadt  
Germany

Phone +49 (0) 6151 869-213

Fax +49 (0) 6151 869-224

E-Mail [info@sit.fraunhofer.de](mailto:info@sit.fraunhofer.de)

URL [www.sit.fraunhofer.de](http://www.sit.fraunhofer.de)

Ed. Michael Waidner

SIT Technical Reports

SIT-TR-2012-001: On the Security of Cloud Storage Services

Moritz Borgmann, Tobias Hahn, Michael Herfert, Thomas Kunz, Marcel Richter, Ursula Viebeg, Sven Vowé

ISBN 978-3-8396-0391-8

ISSN 2192-8169

Printing:

Mediendienstleistungen des

Fraunhofer-Informationszentrum Raum und Bau IRB, Stuttgart

Printed on acid-free and chlorine-free bleached paper.

All rights reserved; no part of this publication may be translated, reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the written permission of the publisher.

Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. The quotation of those designations in whatever way does not imply the conclusion that the use of those designations is legal without the consent of the owner of the trademark.

© by FRAUNHOFER VERLAG, 2012

Fraunhofer Information-Centre for Regional Planning and Building Construction IRB

P.O. Box 80 04 69, D-70504 Stuttgart

Nobelstrasse 12, D-70569 Stuttgart

Phone +49 (0) 7 11/9 70-25 00

Fax +49 (0) 7 11/9 70-25 08

E-Mail [verlag@fraunhofer.de](mailto:verlag@fraunhofer.de)

URL <http://verlag.fraunhofer.de>

# CONTENTS

<b>Executive Summary</b>	<b>10</b>
<b>1 Introduction</b>	<b>13</b>
1.1 Scope	14
1.2 Definition of Cloud Storage Services	15
1.3 Use Cases for Cloud Storage Services	16
<b>Part I: Principles of Cloud Storage Services</b>	<b>18</b>
<b>2 The User's View</b>	<b>19</b>
2.1 Features	19
2.1.1 Copy	19
2.1.2 Backup	20
2.1.3 Synchronization	20
2.1.4 Sharing	21
2.2 Interfaces	21
2.2.1 Proprietary Software Clients	21
2.2.2 Browser Interface	21
2.2.3 Application Programming Interface	21
2.3 Optimization	22
2.3.1 Deduplication	22
2.3.2 Delta Encoding	23
2.3.3 Compression	23
<b>3 The Lawyer's View</b>	<b>24</b>
3.1 Legal Regulations in Germany	24
3.1.1 Data Protection	24
3.1.2 Further Legal Provisions	26
3.1.3 Certification and Guidelines	28
3.2 Legal Regulations in the EU	31
3.2.1 The Data Protection Directive	31
3.2.2 The Safe Harbor Framework	32
3.2.3 Recommendations	33

3.3	Legal Regulations in the USA . . . . .	34
3.3.1	The Patriot Act . . . . .	34
3.3.2	The Fourth Amendment . . . . .	34
<b>4</b>	<b>The Security Engineer's View</b>	<b>36</b>
4.1	Registration and Login . . . . .	36
4.2	Transport Security . . . . .	38
4.3	Encryption . . . . .	39
4.4	File Sharing . . . . .	40
4.5	Deduplication . . . . .	41
4.6	Multiple Devices . . . . .	43
4.7	Update Functionality . . . . .	44
4.8	Server Location . . . . .	44
4.9	Classification of Security Requirements . . . . .	44
4.10	Further Threats . . . . .	47
4.10.1	Time Related Aspects . . . . .	47
4.10.2	Advanced Persistent Threats . . . . .	48
4.10.3	A Note on Client-side Encryption . . . . .	49
<b>Part II:</b>		
<b>Analysis of Cloud Storage Services</b>		<b>51</b>
<b>5</b>	<b>Methodology for Analysis</b>	<b>52</b>
5.1	Selection of Products . . . . .	52
5.2	Scope . . . . .	53
5.3	Format . . . . .	54
<b>6</b>	<b>CloudMe</b>	<b>56</b>
6.1	Synopsis . . . . .	56
6.2	Availability . . . . .	56
6.3	Features . . . . .	57
6.4	Security . . . . .	60
<b>7</b>	<b>CrashPlan</b>	<b>64</b>
7.1	Synopsis . . . . .	64
7.2	Availability . . . . .	64
7.3	Features . . . . .	66

7.4	Security . . . . .	67
<b>8</b>	<b>Dropbox</b>	<b>72</b>
8.1	Synopsis . . . . .	72
8.2	Availability . . . . .	72
8.3	Features . . . . .	72
8.4	Security . . . . .	74
<b>9</b>	<b>Mozy</b>	<b>79</b>
9.1	Synopsis . . . . .	79
9.2	Availability . . . . .	79
9.3	Features . . . . .	80
9.4	Security . . . . .	81
<b>10</b>	<b>TeamDrive</b>	<b>86</b>
10.1	Synopsis . . . . .	86
10.2	Availability . . . . .	86
10.3	Features . . . . .	88
10.4	Security . . . . .	90
<b>11</b>	<b>Ubuntu One</b>	<b>94</b>
11.1	Synopsis . . . . .	94
11.2	Availability . . . . .	94
11.3	Features . . . . .	97
11.4	Security . . . . .	99
<b>12</b>	<b>Wuala</b>	<b>103</b>
12.1	Synopsis . . . . .	103
12.2	Availability . . . . .	103
12.3	Features . . . . .	104
12.4	Security . . . . .	109
<b>13</b>	<b>Summary of Findings</b>	<b>114</b>
<b>Part III:</b>		
<b>Recommendations and Conclusion</b>		<b>115</b>
<b>14</b>	<b>Local Encryption Methods</b>	<b>116</b>

14.1 Approaches enhancing security . . . . .	116
14.2 Approaches not enhancing security . . . . .	119
<b>15 Selecting a Cloud Storage Service</b>	<b>121</b>
<b>16 Conclusion</b>	<b>124</b>
<b>Glossary</b>	<b>127</b>
<b>Acknowledgements</b>	<b>128</b>
<b>References</b>	<b>129</b>
<b>A Attack on CloudMe Desktop</b>	<b>132</b>
<b>B Incrimination attack on CloudMe, Dropbox and Wuala</b>	<b>133</b>



# On the Security of Cloud Storage Services

Moritz Borgmann, Tobias Hahn, Michael Herfert, Thomas Kunz, Marcel Richter,  
Ursula Viebeg, Sven Vowé

Fraunhofer-Institute for Secure Information Technology SIT, Darmstadt

---

Key Words: Cloud Computing, Cloud Storage, Security, Privacy, Encryption, Confidentiality,  
Outsourcing

---

## Executive Summary

The ever-increasing amount of valuable digital data both at home and in business needs to be protected, since its irrevocable loss is unacceptable. Cloud storage services promise to be a solution for this problem. In recent years, their popularity has increased dramatically. They offer user-friendly, easily accessible and cost-saving ways to store and automatically back up arbitrary data, as well as data sharing between users and synchronization of multiple devices.

However, individuals and especially businesses hesitate to entrust their data to cloud storage services since they fear that they will lose control over it. Recent successful attacks on cloud storage providers have exacerbated these concerns. The providers are trying to alleviate the situation and have taken measures to keep their customers' data secure.

In this study we have examined the security mechanisms of seven cloud storage services:

CloudMe, CrashPlan, Dropbox, Mozy, TeamDrive, Ubuntu One, Wuala.

The study may be useful for users of the examined services, but also for users of other services by checking if these services match the identified security requirements.

**Approach** Each service includes a piece of client software and a server-side software. We examined the client software for PCs, never made a penetration test on the server-side. The observation period started in summer 2011 and ended in January 2012.

In a first step, we identified four typical features of cloud storage services. (i) The *copy* feature. This means a service just mirrors a part of the local disk in the cloud. If local hardware drops out (e.g. a stolen laptop) data can be recovered from the cloud. (ii) The *backup* feature which is used to preserve any version of a file in the cloud. (iii) The *synchronization* feature which enables a user to synchronize all of his devices (desktop, laptop, tablet, mobile phone). (iv) The *file sharing* feature which is used for collaboration with project partners. In addition, we have identified optimization features like *deduplication* (files that are known by the server are not transferred again) which may be supported by the service. Each service supports one or more of the features above as shown in table I.

	Copy	Backup	Sync.	Sharing
CloudMe	✓			✓
CrashPlan		✓		
Dropbox	✓		✓	✓
Mozy		✓		
TeamDrive	✓	✓	✓	✓
Ubuntu One	✓		✓	✓
Wuala	✓	✓	✓	✓

Table I. Features of cloud storage services.

Second, we identified security requirements. The top five requirements and their objectives are: (i) *Registration and Login*, to protect against incrimination, information gathering and to enforce usage of strong passwords. (ii) *Transport Security*, to secure communication between client and server. (iii) *Encryption*, to disable the provider to examine stored data. (iv) *Secure File Sharing*, to protect documents shared by a closed group, optionally including non-subscribers. (v) *Secure Deduplication*, to avoid privacy problems when using deduplication.

**Results** We have applied the security requirements to the selected services, as shown in table II.

	Registration	Transport	Encryption	Sharing	Deduplication
CloudMe	--	--	--	-	%
CrashPlan	+	±	+	%	+
Dropbox	-	+	-	±	+
Mozy	±	+	±	%	-
TeamDrive	±	±	+	±	%
Ubuntu One	++	+	--	++	+
Wuala	-	±	±	±	-

Table II. Grades. ++ very good, + good, ± some weaknesses, - bad, -- very bad, % not available

*Registration* was a problem for CloudMe, Dropbox and Wuala because they missed to verify the email address of a new customer. Hence, an *incrimination attack* is possible, that means a person *A* can register with the email address of another person *B*. Now, *A* can upload illegal material using the account of the victim *B*. After that *A* can notify authorities, e.g. the police, about the illegal content.

*Transport Security* was a problem for CrashPlan, TeamDrive and Wuala because they deny the usage of SSL/TLS. Instead they use unpublished, self-made protocols – a very error-prone approach. CloudMe does not take any measure to protect the security of files during transmission.

*Encryption* was a problem for CloudMe, Dropbox and Ubuntu One because they do not use client-side encryption, thus the provider is able to read the data. Mozy does not encrypt filenames. The convergent encryption scheme used by Wuala enables attacks by a server-side attacker.

*Sharing of data* was a problem for CloudMe, Dropbox, TeamDrive and Wuala. Problems occur if files are shared with non-subscribers on the principle of a long, unpredictable URL. CloudMe does not obfuscate this URL adequately. Dropbox gives an unclear description wrt to sharing details, TeamDrive is weak when disinviting a group member and Wuala enables information

gathering by including the user name in public URLs. CloudMe does not prevent search engines from accessing the workspace.

*Deduplication* was a problem for Mozy and Wuala, because in some cases it is possible to ask the cloud storage provider whether a file is already stored or not.

Data confidentiality can be improved by users by encrypting their data locally before uploading it to the cloud. This can be done using a variety of available encryption tools, including TrueCrypt, EncFS and GnuPrivacyGuard. Under some circumstances these tools will interfere with features of the storage service. Users should be conscious that in any case they trust the provider by using client software supplied by the provider.

**Legal Considerations** In addition, we considered legal requirements for a compliant usage of cloud storage services. An examination of laws and legal provisions shows that the cloud user is primarily responsible for his data and its processing. Especially companies have to consider that the legal requirements to which they must adhere may differ to those for the cloud provider or a potential subcontractor. The absence of international regulations guaranteeing an adequate level of data security and privacy requires that European companies must choose cloud providers based within the European Economic Area (EEA). Additionally, these providers should not be subsidiaries of companies based in the United States, otherwise the Patriot Act can be used to gain access to the data stored at the provider, even if it is exclusively stored within the EEA.

**Summary** Individuals or companies considering to use cloud storage services are advised to check whether a cloud provider meets these security requirements.

In addition, it is worthwhile to consider using more than one service to reduce the impacts of service downtime. Further, calculation of the time to recover all data from the cloud is recommended. Depending on the individual amount of data, this may take several days. Having a plan for a provider change in the future reduces the dependancy on a particular provider (*provider lock-in*). This will be relevant, for example, if the chosen provider is getting to expensive or is not longer compliant with governmental rules.

As a major result, the study shows that most of the analyzed cloud storage providers are aware of the extreme importance of data security and privacy, hence they have taken protection measures. However, a solution which meets all of the mandatory security requirements has not been found with any of the analyzed providers.

We hope that this study helps to enhance the security of cloud storage services.

# 1. INTRODUCTION

In recent years, the popularity of cloud storage services has increased dramatically. For instance, the popular service Dropbox surpassed 25 million registered users at the beginning of 2011<sup>1</sup>. Ubuntu One has reached more than one million registered users in July 2011<sup>2</sup> as well as Mozy<sup>3</sup>. These services are used to store the huge amount of digital data which is accumulated in both private and business sectors. Individuals own ever-increasing collections of digital photographs, videos, music (MP3 files), and e-books. Most business processes have been digitalized, i.e., information such as communication data, accounts, contracts, advertising material, construction or business plans only exists in digital form.

The data is often of great value and its irrecoverable loss or damage could be a total disaster for its owner. For parents, videos of their children growing up may be very important, PhD students may rely on digital material, e.g., a collection of Internet references, to be used for a dissertation. For a company, the loss of data could ruin the basis for business. Additionally, companies are legally obliged to preserve tax records for a certain period (6 or 10 years), and to leave them available to the fiscal authorities.

This requires secure methods of preserving important data in order to prevent unrecoverable data loss, whilst constantly keeping up with increasing demands for storage space. It is necessary to regularly make extra copies of the information, so as to be able to restore it to an earlier version if need be. These copies further escalate the demand for storage space. Additional requirements arise from the variety of devices used to access the data simultaneously. Private and business users demand an easy way to synchronize and access their data independent of both device and location. The software providing these features must also be tailored to the needs of the individual with no technical background.

In order to meet these demands, companies make large investments into their IT infrastructure. Additional hardware and software is required, as well as staff for its operation and maintenance. Larger companies might have to consider building a dedicated data center. These expenses conflict with the continuing need to reduce costs in order to stay competitive.

Cloud storage services offer user-friendly, easily accessible and money-saving ways of storing and automatically backing up arbitrary data. These services are available on-demand on the Internet. A customer simply accesses the website of a cloud storage provider and rents storage space as necessary by selecting one of the provider's packages.

---

<sup>1</sup><https://www.dropbox.com/press/20110418>

<sup>2</sup><http://voices.canonical.com/ubuntuone/?p=1023>

<sup>3</sup><http://www.emc.com/about/news/press/2010/20100518-01.htm>

A precondition for using this service is Internet access from the customer's computers or mobile devices. Depending on the amount of data to be transmitted to the cloud, sufficient bandwidth must be available, otherwise the transfer could be very time-consuming. For individuals — depending on their location — this may be a problem, but the availability of reasonably-priced broadband is increasing.

If the use of cloud storage services carries such great advantages, why are individuals and companies alike still hesitant to entrust their data to the cloud?

Usage of a cloud storage provider basically means entrusting data to a third party where no prior relationship based on trust has been established. Individuals who upload personal information to the cloud want to be sure that only certain people are able to access it. This should also exclude the provider, since there is no justifiable reason for it to access the data.

Companies may entrust files containing sensitive business data and valuable intellectual property which may be of great interest for industrial espionage. The unauthorized disclosure of customer information, business secrets or research data poses a serious threat to a company's business. In addition, compliance requirements with both internal security guidelines and legal regulations have to be met. The cloud storage provider may be subject to different legal regulations than the user. The possibility of the cloud provider going out of business needs to be taken into account, since the data might not be easily transferable to another provider (“vendor lock-in”).

Recent incidents (e.g. [MSL<sup>+</sup>11], [New11]) where the vulnerabilities of cloud storage providers have been exploited show that doubts concerning their usage are justified.

## 1.1 Scope

The main challenge of cloud storage is guaranteeing control, and the necessary integrity and confidentiality of all stored data.

This study's intended readership is those companies and individuals interested in or planning to use cloud storage services. It aims to sensitize users to existing privacy, security and legal issues.

In Section 2 the features of cloud storage services will be described from a user's point of view. Legal aspects and implications for privacy will be discussed in Section 3 from a lawyer's point of view. Cloud storage providers should meet a set of security requirements, which will be defined and described within section 4 from a security engineer's point of view.

The analysis in part 2 starts in Section 5 with a description of our methodology. The requirements will be used to analyze a few selected cloud storage providers in sections 6 to 12. The analysis includes relevant organizational, legal and technical information for each service. Section 13 subsumes the found security weaknesses.

Part 3, recommendations and conclusion, begins with an overview on the methods and tools available for achieving service-independent security of personal data using local encryption methods in section 14. After that, Section 15 gives an assistance to select a cloud storage service. The study closes with a conclusion.

The appendices describe attacks made possible through missing security features of the cloud storage providers. All service providers have been informed prior to publication of this study.

The text assumes some basic knowledge about information technology in general and, in particular, cloud computing. The National Institute of Standards and Technology (NIST) provides a good definition which introduces all basic terms concerning cloud computing [MG11]. Several technical terms from the field of IT-Security are explained in the glossary at the end of the study. These entries are set in *italic* and followed by an up arrow (*glossary entry*↑).

The study makes no claim to be complete. It is limited to the analysis of only a few advanced cloud storage services offering easy to use client interfaces, and excludes basic cloud storage services like Amazon S3. Furthermore, no ranking is attempted because the user requirements are very different.

We make some security assumptions in order to focus on cloud storage services. In particular: We assume, the client's computer is free of any malware. We assume, correct implementations of cryptographic algorithms on the client's computer and we assume, the client is the only one having control of is mail account.

Last but not least, the study has been written by computer scientists with the focus on IT security. The legal considerations in this paper are not made to substitute an advisory by a lawyer.

## 1.2 Definition of Cloud Storage Services

Basically, a cloud storage system can be considered to be a network of distributed data centers which typically uses cloud computing technologies like virtualization, and offers some kind of interface for storing data. To increase the availability of the data, it may be redundantly stored at different locations. In general, all of this is not visible to the user.

Many cloud storage providers are active on the market, offering various kinds of services to their customers. This study distinguishes between two types of cloud storage services:

**Basic cloud storage services** are generally not designed to be accessed directly by users but rather incorporated into custom software using *application programming interfaces*↑ (API). Ex-

amples of such basic cloud storage services are Amazon S3<sup>4</sup>, Rackspace<sup>5</sup> and Nirvanix<sup>6</sup>.

**Advanced cloud storage services** mostly employ basic cloud storage services for the actual storage of data, and provide interfaces such as client or web applications which greatly simplify the use of the service for the customer. Many services may also provide an easy to use API to allow integration of the service's capabilities into third-party software. Examples of advanced cloud storage services are Dropbox<sup>7</sup>, and Mozy<sup>8</sup>.

### 1.3 Use Cases for Cloud Storage Services

There are multiple use cases for cloud storage services used by both companies and individuals. This includes on-demand storage capacities accessible from various locations (e.g. from mobile and local devices), backup facilities without the need to maintain hardware devices or appropriate software tools, and synchronization features allowing the customers to always have access to the latest version of their data independent of the device (PC, laptop, smartphone). The following use cases shall present the potential benefits of such storage services.

- (1) *Copy*. Bob is a sales representative, working most of the time out of the office. At the moment he is traveling home from a two weeks business trip. He has visited many customers, made a lot of notices and sketched some drafts for potential missions. His trip was very successful, but Bob is depressed. As he was inattentive while waiting for the plane, the bag including his laptop has been stolen. He does not worry about disclosure of data because he uses an encrypted harddisk, but he is sad when he is thinking about the results of the last two weeks. Using the *copy* feature of a cloud storage service Bob would be able to solve this problem. His laptop continuously copies all changes to existing data and all new files to the cloud service. Back home, it is convenient to recover this data to his new laptop.
- (2) *Backup*. Charlie, an architect, is the owner of a medium sized and prosperous architectural office. This day, he is desperately seeking a building plan which he had drawn up last year. It is the house plan for a customer who has repeatedly changed his mind, forcing Charlie to frequently alter it. Yesterday, the customer was in Charlie's office and demanded a redesign based on a plan from March 5th, 2011. The customer presented a paper copy of this former version. Since then the architect has been searching for the digital version. Although he

---

<sup>4</sup><http://aws.amazon.com/s3>

<sup>5</sup><http://www.rackspace.com/>

<sup>6</sup><https://www.nirvanix.com/>

<sup>7</sup><http://www.dropbox.com>

<sup>8</sup><http://www.mozy.com>



protects his data by making copies and storing them on external devices, he cannot find that particular version.

Using a reliable *cloud backup service*, which keeps all versions of all files, would have enabled Charlie to restore the building plan from any particular date.

- (3) *Synchronization of Devices*. Diana has worked the first half of the day in the office on a presentation for tomorrow. At the moment she is traveling to the location of her speech. She has planned to polish her slides in the train. Unfortunately, she worked in the office on her Desktop PC and she forgot to copy the presentation on an USB stick.

Using a service that offers a *synchronization* feature Diana could solve this problem. Her desktop PC would permanently copy all data to the cloud, like in the storage scenario above. In addition the synchronization feature allows the connection of any number of devices. The service takes care that all devices have the same data pool. Even if she creates new files on different devices while she is offline the service recognizes that. Even better, if she changes one file on two devices, the service is able to detect version conflicts.

- (4) *Sharing Files*. Eve is a self-employed photographer. At the moment she is writing a travel guide, together with Frank, a talented writer. They are working together using email. Both of them are displeased on this scenario because the book is made of more than 100 files and it is very cumbersome to send them via email and to hold them in sync.

Using a service that offers a *file sharing* feature working on any number of files would be much easier. Eve and Frank could just edit their piece of work and each computer would copy the results immediately to the cloud. Being there, the other computer would automatically download the file. So both of them are always in sync, without any pain. And if they accidentally edit the same file at the same time, the service creates a notice and offers alternatives how to proceed.

# PART I: PRINCIPLES OF CLOUD STORAGE SERVICES

## 2. THE USER'S VIEW

This section introduces the typical features of cloud storage services. A particular service must offer at least one of these features, and may offer multiple features at the same time.

### 2.1 Features

In the following, an explanation of the features *copy*, *backup*, *synchronization* and *sharing* will be given (cf. Fig. 1).

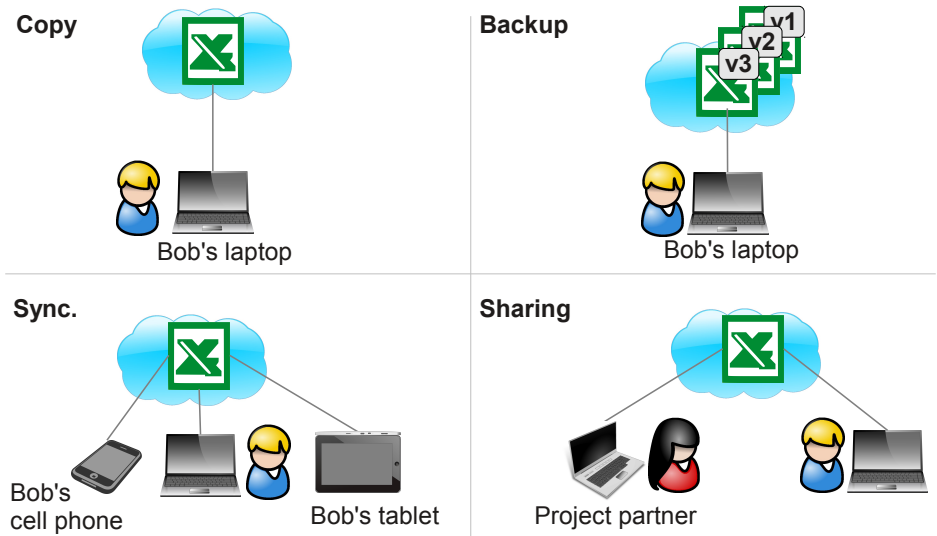


Figure 1. Features of storage services.

#### 2.1.1 Copy

The copy feature creates a mirror of current local data in the cloud. The typical user wants to assure that data is available even if local hardware drops out (e.g. by a hard disk crash or a stolen laptop). Further, he wants to access his data from any place, even if his own hardware is not available. Therefore, an access via web browser is quite usual, for a service providing a copy feature.

In contrast to the backup feature where data is stored at certain times the copy feature usually stores data continuously. A storage service may provide a short retention period, e.g. 30 days, to

recover deleted data but this time is too short to satisfy the definition of a backup service as given above.

Typically, there are different ways to store the data in the cloud. The customer may manually store single files or folders in online storage using his web browser, or he may use client software provided by the cloud provider. Such client software has to be locally installed by the customer and may be used for the automatic upload of, for example, all (new) files from a given folder belonging to the client to the cloud storage.

### 2.1.2 Backup

The backup feature allows to recover any version of a previously stored file or directory over a long period of time, usually many years. The typical user wants to sustain his intellectual property and to fulfill compliance requirements.

Creating backups using cloud services is an automated process of periodically making copies of data, transmitting these copies to and storing them in the cloud so that they may be used to restore the original after a data loss event.

Cloud backup service providers usually offer software to be installed locally, enabling the customer to select the data to be backed up, to configure the retention period as well as a schedule for the backups. The client software either runs continuously in the background so that newly created or changed files are backed up immediately or the software is configured to perform the backup on a regular basis (e.g. hourly). One task of the client software is to check which data needs to be backed up, i.e. to recognize files added or changed since the previous backup.

Additionally, the client software could enable the customer to monitor the backup process, i.e. the customer is able to view the backup and restore history. This might be implemented as a continuously written log file where all actions are recorded.

### 2.1.3 Synchronization

Synchronization is the process of establishing consistency among data from different sources. The typical user has a set of devices, e.g. a laptop, a tablet and a smartphone, and wants to have all data available on all devices and that all data can be changed on every device.

The client software must be able to detect conflicts that occur if a file has been changed on two devices in different ways. The software should offer a number of choices to the user: either merge the files, overwrite one version, or keep both versions by applying a renaming scheme.

### 2.1.4 Sharing

Data sharing is the process of sharing data with (1) other subscribers of the same service, (2) with a closed group of people from the outside, or (3) with everybody. The typical user wants to collaborate with colleagues and project partners or to share data with his friends or to publish data.

Depending on the service, the shared data has a set of fixed or configurable access rights like read, write, upload or delete. If write access is enabled for more than one user synchronization problems as described in the previous section may arise.

## 2.2 Interfaces

This section explains the different interfaces that can be used to access the data at the cloud storage provider.

### 2.2.1 Proprietary Software Clients

The most comfortable way to use a cloud storage service is to install the proprietary client which is distributed by the service provider. Every provider analyzed in this study provides such a client. The client allows the selection of data which should be transmitted to the cloud, the management of the service (e.g. buying more storage) or the configuration of features like sharing or synchronization. Often, clients are available for a couple of operating systems.

### 2.2.2 Browser Interface

A web browser interface is a method to access data from any place even on a device which has no client installed and which is owned by somebody who is not identical to the user who has submitted the data to the cloud. A browser interface is sometimes preferred by companies that do not want to spend too much time to manage software for their employees. Further, it is wanted by private users who appreciate a way to share their data, e.g. photos from the last holidays, wherever they are, even on foreign devices.

### 2.2.3 Application Programming Interface

Most cloud storage providers grant their users access to an *application programming interfaces*<sup>†</sup> (API). This API can be used by developers to directly integrate access to the cloud storage service into applications, e.g., to provide games for a mobile device game across multiple devices and platforms. If the API supports advanced features such as deduplication or access to revisions of files previously saved within the cloud storage, these can be integrated to enhance the capabilities of in-house applications.

In order to provide an API for customers, cloud storage providers need to expose a web service or web application which can be accessed using a standardized communication protocol. The majority of cloud storage providers offering an API either expose standard web services leveraging the SOAP protocol [HNM<sup>+</sup>03] or use the Representational State Transfer [Fie00] in the form of RESTful web services. In order to facilitate cloud storage API usage, developers are generally provided with *software development kits*<sup>†</sup> (SDK).

## 2.3 Optimization

This section explains the optimization techniques *deduplication*, *delta encoding* and *compression*, which are provided by some services in order to save bandwidth.

### 2.3.1 Deduplication

The term *Deduplication* (also *Data Deduplication*) describes a popular technique that allows cloud storage providers to significantly decrease the amount of needed storage space. The principle of deduplication is as follows: only a single copy of each piece of data is stored. If a user wants to store data that the cloud storage provider already has stored in the past, the storage provider simply creates a link to that data instead of storing another copy. There are some variations of how deduplication may be realized:

- (1) *File level deduplication vs. block level deduplication.* File level deduplication means that only a single copy of each file will be stored. Block level deduplication means that each file will be split up into blocks and only a single copy of each block will be stored. Identical files or blocks are detected by comparing the *hash value*<sup>†</sup> with a list of known files or blocks.
- (2) *Server-side deduplication vs. client-side deduplication.* In the case of server-side deduplication, each file a user wants to store is transmitted to the cloud storage provider. For every file, the provider checks if he has to store the file or only needs to create a link to an already stored file. The user cannot detect if the cloud storage provider uses data deduplication. In the case of deduplication by the client, the client software transmits the hash value of the file to the cloud storage provider. Only if the provider is not already in possession of the file it will be transmitted. This variation of data deduplication has the effect of not only saving storage space, but also bandwidth. It is easy to detect if a cloud storage provider uses this kind of data deduplication by inspecting the log files or observing the amount of data that is transferred.
- (3) *Single user deduplication vs. cross user deduplication.* Single user deduplication means that data deduplication is carried out separately for each user: If user *A* wants to store a file he has already stored in the past or in a different folder, the cloud storage provider only creates a link to that file. In the case of cross user deduplication, data deduplication is carried out across all

users: If user *A* wants to store a file that another user *B* has already stored, the cloud storage provider only creates a link to that file instead of storing an additional copy.

In general, data deduplication is carried out completely in the background which means that the user usually cannot choose whether data deduplication should be used or not. Deduplication may cause security problems, which will be discussed in Section 4.5.

### 2.3.2 Delta Encoding

A popular technique for minimizing data transfer and thus saving bandwidth is to only upload the differences to a previously uploaded file instead of transferring the whole file. Suppose that a user wants to store a file that only slightly differs from a previously stored file. In this case, it is not necessary to upload the entire file. Instead, it is sufficient to only upload those parts of the file differing from the previous version along with additional information needed to reconstruct the file on the server. A common method to implement this kind of optimization is the algorithm used by the `rsync`<sup>9</sup> tool: The principle is to split a file into chunks of fixed size. For each chunk, it is checked whether the cloud storage provider has already stored this chunk or not (e.g. by sending a *hash value*<sup>†</sup> of the chunk). Only those chunks that do not match any chunk already stored by the cloud storage provider will be uploaded.

Delta encoding does not make sense if the service cannot decrypt data because two cryptograms of slightly modified inputs may differ completely.

### 2.3.3 Compression

A simple way to save bandwidth is to compress data on client-side. A drawback is that compression consumes computing power, which may cause trouble to users of storage services where transmission of data to the cloud is a continuous process. Compression can be combined with delta encoding and works fine with encrypted data, if applied before encryption.

---

<sup>9</sup><http://rsync.samba.org/>

### 3. THE LAWYER'S VIEW

Laws and legal provisions regulating the use, processing and storage of data have to be adhered to when storing data in the cloud. This applies to all parties involved: the cloud user who stores data in the cloud, the cloud provider who offers cloud storage services, and potential subcontractors who provide resources or infrastructures for the cloud provider. In this context, a cloud user may be an individual or a business company. Each of the aforementioned parties is subject to legal regulations and provisions of the country in which the respective party is based. These legal regulations and provisions may concern personal rights (e.g. data privacy), data security or access rights for fiscal or law enforcement authorities and may differ from country to country. In general, the cloud user is primarily responsible for his data and its processing. Therefore, a cloud user choosing a cloud storage provider has to consider that the legal requirements to which he must adhere may differ to those for the cloud provider or the subcontractor.

In the following, we will examine the impact on the legal requirements to which a cloud user must adhere.

#### 3.1 Legal Regulations in Germany

In Germany several legal provisions and laws regulate the use, processing and archiving of data. Those legal requirements must be adhered to by cloud users when storing data in the cloud. Which regulations have to be followed when using cloud computing services depends on the kind of information being stored in the cloud. Data which concerns the personal rights of others, so-called personal data, has to be protected according to the German Federal Data Protection Act<sup>10</sup> (Bundesdatenschutzgesetz (BDSG) [BfD10]). The cloud user (an individual or a business company) is responsible to ensure data protection when storing personal data in the cloud. There is no legal regulation for individuals storing data which only concerns themselves. Companies are, in addition, subject to further legal regulations regardless whether they are using cloud storage or not. The following sections analyze which legal regulations must be adhered to by cloud users.

##### 3.1.1 Data Protection

In Germany, the Federal Data Protection Act (BDSG) regulates the handling of personal data in order to protect individuals against infringement of their right to privacy. It applies to the collection, processing and use of personal data, which includes, according to § 3(1) BDSG

---

<sup>10</sup>[http://www.bfdi.bund.de/cln\\_111/EN/DataProtectionActs/DataProtectionActs\\_node.html](http://www.bfdi.bund.de/cln_111/EN/DataProtectionActs/DataProtectionActs_node.html)



“... any information concerning the personal or material circumstances of an identified or identifiable natural person (data subject)”.

Personal data may be any information concerning a company’s employees, business partners, suppliers or customers or any other information which can be (directly or indirectly) attributed to a natural person.

In practice, the data protection law will apply to almost every business application, unless the collected data is completely anonymized. Pseudonymization, in § 3(7) BDSG named as ‘aliasing’, may be a method to achieve a level of privacy sufficient to permit data processing as well [Wei10].

The BDSG also applies to personal data gathered, processed or used in the cloud. The responsible entity to guarantee the protection of the personal data is defined in § 3(7) BDSG as the controller:

“ ‘Controller’ shall mean any person or body which collects, processes or uses personal data on his, his or its own behalf, or which commissions others to do the same.”

In cloud computing the controller is the cloud user, i.e., the companies or individuals which store data in the cloud.

One possibility to achieve the protection of the personal data is to use a private cloud<sup>11</sup>, where all devices are under complete control and legal responsibility of the controller (cloud user). In this case the cloud user himself has to take all necessary measures to ensure data security and data privacy according to BDSG. The disadvantage of using a private cloud is that the cloud user has to buy, build and manage the hardware and software him and thus may not benefit as much from the advantages cloud computing offers as when using a public cloud. Using a private cloud may be an alternative only for companies with an adequate IT team managing it.

If the cloud user chooses a public cloud service, he commissions someone else (the cloud provider) to store his data, with the result that the cloud user loses control to some extent over his data and its safety. According to § 3(7) BDSG cloud computing can legally be seen as *Contract Data Processing* (in German ‘Auftragsdatenverarbeitung’), which is regulated by § 11 BDSG. A company delegating the processing of personal data to a third party – the cloud provider – is required to ensure that the third party complies with the BDSG. This includes the following topics:

- Careful selection of the provider, including a written contract that includes all items from § 11(2) BDSG, which comprises, for example, the following articles:
  - the duration of contract work
  - data security measures to be taken according to § 9 BDSG
  - the provider’s (monitoring) obligations

---

<sup>11</sup>A definition of private and public clouds can be found in [MG11]

- any right to issue subcontracts
  - the cloud user's monitoring rights
  - the return of data storage media and the deletion of data recorded by the provider after the work has been carried out
- Making sure that personal data is not transferred outside the European Economic Area (EEA) according to § 4b BDSG.
  - Making sure that the data processor provides an adequate level of protection, i.e. the cloud user shall verify compliance with the technical and organizational measures taken by the cloud provider before data storing begins and regularly thereafter. (§ 11 BDSG).

According to § 11(2) no. 6 BDSG the above mentioned items apply not only to the cloud provider as the prime contractor but also to potential basic cloud storage providers, which may act as sub-contractors [Wei10].

*Exception:* Cloud computing outside the EEA is legally not regarded as 'contractual data processing' as mentioned above. If the personal data (in the cloud) is not processed in Germany, in another European Union Member State, or another state party to the Agreement on the EEA (§ 3(8) BDSG), cloud computing is legally seen as a *data transfer* (§ 4b BDSG). A data transfer outside the EEA usually requires a separate legal admission or additional contractual regulations (e.g. so-called binding corporate rules (BCR)) to ensure an adequate privacy level according to § 4c(2) BDSG. Such BCRs must be approved by the responsible data protection authorities [Wei10].

### 3.1.2 Further Legal Provisions

Besides the data protection requirements, further legal provisions concerning retention of data or fiscal requirements have to be met by a company when using cloud computing. The following list contains some examples of regulations with which a German company has to comply:

- Handelsgesetzbuch (HGB [BMF11b]; in English: Commercial Code<sup>12</sup>) regulates the legal obligation of a company to keep and preserve accounting records (§ 238 HGB). Central requirements concern the traceability of tax-relevant business processes. According to § 239 (2) HGB and to the GoBS (Grundsätze ordnungsgemäßer datenverarbeitungsgestützter Buchführungssysteme [AWV95]; in English: Principles of orderly Computer-Assisted Accounting Systems<sup>13</sup>) the preservation process must ensure the completeness, integrity, authenticity, and availability of the tax-relevant accounting data. Additionally, § 257 HGB determines which

---

<sup>12</sup>[http://www.archive.org/stream/germancommercial00germuoft/germancommercial00germuoft\\_djvu.txt](http://www.archive.org/stream/germancommercial00germuoft/germancommercial00germuoft_djvu.txt)

<sup>13</sup>May be ordered under <http://www.awv-net.de/schriften/I-09546-e.html>

documents (e.g. accounting documents and business letters) need to be preserved and it defines a retention period of 6 or 10 years.

A company which stores tax-relevant data in the cloud must ensure that the commissioned cloud provider meets all the requirements concerning retention period, safety, and accessibility of the stored data. In general, the cloud user has to make a contract with the cloud provider which contains all necessary provisions e.g. governing the liability of the cloud provider and potential cloud subcontractors towards the cloud user.

- AO (Abgabenordnung [BMF11a]; in English: Fiscal Code of Germany<sup>14</sup>) regulates in which location (country) tax-relevant data may be stored and under which conditions such data may be stored in a European Union (EU) or European Economic Area (EEA) member state (§ 146 (2a) AO). According to § 148 AO, the responsible tax authority may allow storing data outside the EU/EEA countries, if storage in Germany would be a hardship for the taxpayer and taxation will not be hindered. Additionally, § 147 AO requires a retention period of 6 or 10 years for tax-relevant data.

When choosing a cloud provider for the storage of tax-relevant data, a company must take into account the location (country) where a cloud provider and potential subcontractors store the data.

- GDPdU (Grundsätze zum Datenzugriff und zur Prüfbarkeit digitaler Unterlagen [BMF01]; in English: Principles of data access and auditing of digital documents<sup>15</sup>) regulates the right of the fiscal authorities to access electronically created tax-relevant data like electronic billings within the scope of tax audits in accordance with § 147 AO. The taxpayer is legally obliged to ensure access to electronically created tax-relevant data for the fiscal authorities.

When storing tax-relevant data in a cloud, a company must choose a cloud provider which ensures integrity and availability of the company's data.

The use of cloud storage in the financial, insurance, telecommunication, or health care sectors may have further (legal) implications as these sectors have their own specific rules and provisions. Outlining these, however, is beyond the scope of this study. In general, the legal implications and problems when processing (personal) data in the cloud are not yet sufficiently addressed and solved. [Wei10] recommends

“With international regulations in place, it would doubtless be possible to make cloud-based data processing independent of location, and to mandate that cloud-based data processing be governed exclusively by the law that applies to the user or to the cloud provider in direct contractual relationship with the user. So far, however, there is no

---

<sup>14</sup>[http://www.gesetze-im-internet.de/englisch\\_ao/index.html](http://www.gesetze-im-internet.de/englisch_ao/index.html)

<sup>15</sup><http://www.avendata.de/downloads/e-GDPdU.pdf>

evidence of efforts in this direction. Given the inconsistency and in some cases the inadequacy or total absence of national laws on data processing in general and data privacy in particular, it is unrealistic to expect international standards for the moment. As a result, we have no alternative but to implement a clear system of legal protections that begins with the data controller, i.e., the cloud user.”

### 3.1.3 Certification and Guidelines

Besides the problem of closing a written contract in a dynamic cloud environment, cloud users, especially small and medium sized companies, often lack the resources and the experience to make the right decisions when choosing a cloud provider, to understand all legal implications when using cloud services, and to be able to assess cloud providers. A solution for these problems may be the use of certified cloud services and the study of already existing guidelines concerning cloud computing. Some of the cloud storage providers advertise with a certification. However, users often cannot evaluate the relevance of such certifications. The following list provides some certification entities and guidelines:

**SAS 70** (Statement on Auditing Standard No. 70 [AIC09]) is an audit guide, whose latest version has been issued by the American Institute of Certified Public Accountants (AICPA) in May 2009. An analogous German standard “IDW PS 951” [IDW10] based on SAS 70 has been issued by the German “Institut der deutschen Wirtschaftsprüfer”. SAS 70 is a guideline for service auditors that assess internal controls of service organizations or service providers, e.g. cloud providers. The audit can only be performed by an independent certified public accountant (CPA). SAS 70 is not a pre-determined set of standards that a service provider must meet to “pass”, i.e. the evaluation criteria can be customized. The (cloud) service provider itself is responsible for describing the controls that will be disclosed in the service auditor’s report. The control may refer to application development, maintenance, security and access, data processing or business continuity. In the USA, SAS 70 has grown increasingly popular with the implementation of the Sarbanes Oxley Act (SOX), an US federal act, that requires an annual report on the effectiveness of internal control over financial reporting of companies listed on the stock exchange.

At the end of a SAS 70 audit, the service auditor issues a report including his opinion on whether the controls were suitably designed, placed in operation, and operating effectively. There are two types of Service Auditor’s Reports:

- A Type I report contains the service organization’s description of controls at a specific point in time (e.g. June 30, 2011).
- A Type II report includes a Type I report and additionally a description of detailed testing of the (cloud) service provider’s controls over a minimum six month period (e.g. January 1, 2011 to June 30, 2011) and the results of those tests.

A SAS 70 report has no validity period. However, most service providers will have the SAS 70 audit conducted annually.

In June 2011, SAS 70 has been superseded by SSAE 16 (“Statement on Standards for Attestation Engagements No. 16, Reporting on controls at a Service Organisation”) issued by AICPA. Additional requirements, e.g., a binding statement of the business management, and some other changes shall increase the quality, informative value, and the binding character of the audit report.

*Conclusion:*

Internationally working companies that have to be compliant to SOX should look for a cloud provider with SAS 70 certification. Generally speaking, the SAS 70 report provides the information that the respective provider has implemented at least a minimum set of quality standards. However, SAS 70 reports cannot be used to compare providers as each provider determines what shall be audited. A cloud user who has to meet certain legal requirements has to check the content of the report very carefully. The user should request a copy of the SAS 70 report from the cloud provider to evaluate if the cloud provider meets all of his security and compliance requirements. The new standard SSAE 16 has just been introduced. Therefore, no conclusion can be drawn regarding its relevance.

**ISO 27001** [ISO05] is an international standard issued by the ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) concerning the information security management in all types of organizations. ISO 27001 was published in 2005 and is the replacement for the original British standard (BS) 7799-2. It is intended to provide the foundation for third party audits, and is “harmonized” with other management standards, such as ISO 9001 and ISO 14001.

The standard specifies requirements for the management of IT security in companies. It provides

“a model of establishing, implementing, operating, monitoring, reviewing, maintaining and improving a documented Information Security Management System (ISMS) within the context of the organization’s overall business risks. It specifies requirements for the implementation of security controls customized to the needs of individual organizations. The ISMS is designed to ensure the selection of adequate and proportionate security controls that protect information assets.” [ISO05]

Organizations with such an ISMS can be audited and certified compliant with the standard ISO 27001.

A certification according to ISO 27001 is international, i.e., national accreditation bodies have a mutual recognition model in place, enabling certifications granted in one country to be recognized in another. To meet the certification requirements, an organization’s ISMS must be audited by a “certification body” which has to be accredited by the national accreditation body for the country in question (e.g., BSI in Germany). A certificate is valid for 3 years.

The German BSI<sup>16</sup> (Bundesamt für die Sicherheit in der Informationsrechnologie; in English: Federal Office for Information Security) offers a certification according to ISO 27001 on the basis of the German IT security guidelines (in German: IT-Grundschutz<sup>17</sup>) which comprises both the control of the IT Security Management and the additional assessment of concrete IT security measures according to the German IT security guidelines. This certification may be performed by an auditor accredited by the German BSI. It is valid for 2 years.

*Conclusion:*

An ISO 27001 certificate is internationally accepted and ensures a certain level of quality and security standards. However, it depends on the importance of the data to be stored in the cloud to which extent the cloud user should check the auditor and the audit-report by him. Even a certification according to ISO 27001 on the basis of the German IT security guidelines does not necessarily imply that the respective company fulfills all data protection requirements according to BDSG.

**EuroCloud Germany\_eco e.V.** EuroCloud Germany\_eco e.V.<sup>18</sup>, an association of the German cloud computing industry, which was founded in February 2010, introduced a “Seal of approval” for Software as a Service (SaaS)<sup>19</sup> cloud services. Eurocloud Germany represents the German Cloud Industry in the Pan-European EuroCloud network<sup>20</sup> in order to find international solutions. The overall goal of this organization is to promote SaaS, cloud services and applications across Europe and encourage its distribution. EuroCloud Germany\_eco tries to address the problems of companies considering the use of cloud services. Small and medium-sized companies in particular lack the experience, the resources, and the legal know-how to choose a trustworthy cloud provider, to make a contract covering all security and legal aspects, and to inspect cloud providers. EuroCloud Germany\_eco awards the Seal of Approval for SaaS products which have passed tests of service, data security, data protection, contract terms, and interoperability. Companies intending to be certified may choose the level of certification themselves and hereby set the level of testing.

EuroCloud Germany\_eco announced at the fair CeBIT 2011 that their test criteria have been developed in coordination with the German BSI. The “Security Recommendations for Cloud Computing Providers” [BSI11] issued by the German BSI have been integrated into the test criteria<sup>21</sup>.

---

<sup>16</sup><https://www.bsi.bund.de>

<sup>17</sup><https://www.bsi.bund.de/ContentBSI/grundschutz/zert/ISO27001/Schema/zertifizierungsschema.html>

<sup>18</sup><http://www.eurocloud.de>

<sup>19</sup>A definition of SaaS can be found in [MG11]

<sup>20</sup><http://www.eurocloud.org>

<sup>21</sup><http://www.eurocloud.de/2011/03/03/das-gutesiegel-fur-die-cloud-erste-anbieter-zertifiziert-optivo-und-pironet-ndh-datacenter-erste-auditierte-anbieter-des-eurocloud-star-audit-saas>

Two cloud providers have already successfully passed the certification process. More certification processes are currently being worked on.

Additionally, EuroCloud Germany<sub>eco</sub> has published a guideline [EHG<sup>+</sup>11] concerning the German law, data protection, and compliance. The overview of topics is derived from the test criteria used by the EuroCloud Seal of approval. Among other items, the guideline provides legal requirements to be met when using cloud services and core items to be taken into account when making a contract with a cloud provider.

*Conclusion:*

The EuroCloud SaaS Quality Seal has just been introduced. Therefore, only little experience has been made regarding the relevance of this certification.

## 3.2 Legal Regulations in the EU

### 3.2.1 The Data Protection Directive

In 1995, the EU established Directive 95/46/EC (Data Protection Directive), which had to be integrated into the laws of each nation belonging to the European Economic Area (EEA) by the end of 1998. The directive aims to protect the rights and freedoms of persons and sets strict limits on the collection and use of personal data. It relates to:

- The quality of the data
- The legitimacy of data processing
- Information to be given to the subject
- The right to object to the collection

A summary<sup>22</sup> and the entire directive<sup>23</sup> provide additional information. Following these data protection laws, companies located within the EEA are not allowed to transfer personal data to anyone outside the EEA, unless the recipient can guarantee an “adequate” level of protection.

Fifteen years earlier, the OECD had already addressed the same topic when they issued their “Recommendations of the Council Concerning Guidelines Governing the Protection of Privacy and Trans-Border Flows of Personal Data”. In these Guidelines<sup>24</sup>, they defined a set of seven basic principles for national application which include the following:

- The Use Limitation Principle mandates that personal data should not be disclosed, made available or otherwise be used except by the authority of law or with the consent of the data subject.

---

<sup>22</sup>[http://europa.eu/legislation\\_summaries/information\\_society/data\\_protection/114012\\_en.htm](http://europa.eu/legislation_summaries/information_society/data_protection/114012_en.htm)

<sup>23</sup><http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31995L0046:EN:NOT>

<sup>24</sup>[http://www.oecd.org/document/18/0,2340,en\\_2649\\_34255\\_1815186\\_1\\_1\\_1\\_1,00.html](http://www.oecd.org/document/18/0,2340,en_2649_34255_1815186_1_1_1_1,00.html)

- The Collection Limitation Principle mandates that there should be limits to the collection of personal data.
- The Purpose Specification Principle mandates that data collection be carried out according to a certain purpose which has to be defined in advance to the actual collection.
- The Disclosure Principle mandates that data subjects should be informed as to who is collecting their data.

These guidelines were non-binding and were mostly not integrated into the national laws of the EU states. All seven principles were later on incorporated into the EU directive.

### 3.2.2 The Safe Harbor Framework

The Safe Harbor framework was created in cooperation between the US department of commerce and the EU to ensure the safe passage of data from Europe to the US<sup>25</sup> and was approved by the EU in 2000. Companies that want to be a part of the Safe Harbor agreement have to comply with the rules defined by the agreement<sup>26</sup>. These rules include, among others

- The notification of individuals about the purposes of data collection
- The access for individuals to personal information about them
- The obligation of the company to protect personal information from loss, misuse and unauthorized access

There is no external validation required to join the Agreement, the joining company has to self-certify to the Department of Commerce that it adheres to the Safe Harbor Framework. This has been one of the points criticized ever since the creation of the agreement in 2000. Although the Federal Trade Commission is authorized to enforce the Safe Harbor Framework, there have only been seven cases in the last ten years<sup>27</sup> where the FTC has filed complaints against companies that violated Safe Harbor Principles. [MS11] points out that the FTC also seems to disregard complaints from individuals concerning Safe Harbor violations.

In a recent study [Con08] regarding the current state of the Safe Harbor Framework, all organizations listed on the Safe Harbor List<sup>28</sup> were examined. Only 1109 of the 1597 companies were current members of the Safe Harbor Framework, the rest (called “phantom” members in Figure 2) no longer existed, or failed to renew their certification. 206 organizations claimed on their public

---

<sup>25</sup><http://www.zdnet.com/blog/igeneration/safe-harbor-why-eu-data-needs-protecting-from-us-law/>

<sup>26</sup>[http://export.gov/safeharbor/eu/eg\\_main\\_018476.asp](http://export.gov/safeharbor/eu/eg_main_018476.asp)

<sup>27</sup><http://writ.news.findlaw.com/ramasastry/20091117.html>

<sup>28</sup><https://safeharbor.export.gov/list.aspx>



### Safe Harbour Framework

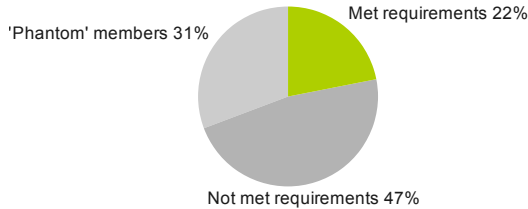


Figure 2. Just 22% of the companies on the Safe Harbour list met the requirements

websites to be current members of the Safe Harbor although they are not. Only 348 organizations met the most basic requirements of the Safe Harbor Framework.

The “Düsseldorfer Kreis”, a German association of regulatory authorities for the enforcement of data protection in the private sector, has published a resolution on the Safe Harbor Agreement<sup>29</sup>. In this resolution, they declare that a European company which wants to transfer personal data to an American company has to verify that the American company correctly implements the Safe Harbor Principles. Relying on self-certification is not sufficient, since currently there is no comprehensive control by the FTC in place.

### 3.2.3 Recommendations

According to [MS11], it can be assumed that it is currently not possible to realize the transfer of personal data to outside of the EEA in accordance with European law. This requires the processing and storage of personal data within the EEA. Taken the Patriot Act (cf. Section 3.3.1) into account, this also excludes any European company that is a subsidiary of an American company. To quote Zack Whittaker<sup>30</sup>

“There is no privacy in the European cloud, or any public cloud outside of the United States where a US-based or wholly owned subsidiary company is involved.”

<sup>29</sup>[http://www.bfdi.bund.de/SharedDocs/Publikationen/Entschliessungssammlung/DuesseldorferKreis/290410\\_SafeHarbor.html](http://www.bfdi.bund.de/SharedDocs/Publikationen/Entschliessungssammlung/DuesseldorferKreis/290410_SafeHarbor.html)

<sup>30</sup><http://www.zdnet.com/blog/igeneration/usa-patriot-act-the-myth-of-a-secure-european-cloud/8807>

## 3.3 Legal Regulations in the USA

### 3.3.1 The Patriot Act

The USA Patriot Act<sup>31</sup> was signed into law in 2001 in response to the terror attacks of September 11th. The case study “How the USA PATRIOT act can be used to access EU data<sup>32</sup>” shows how the Patriot Act could be used in the context of cloud computing. It affects not only data that is stored in the US in accordance with the Safe Harbor Agreement but also data that is physically stored in Europe. The latter case is also relevant when the data is being stored by a European company that is a subsidiary of an American company. Confirmation of this point came directly from Microsoft<sup>33</sup>. A company representative was asked at the launch of their new cloud service Office365 whether Microsoft can

“... guarantee that EU-stored-data, held in EU based data centers, will not leave the European Economic Area under any circumstances — even under a request from the Patriot Act.”

The reply was:

“Microsoft cannot provide these guarantees. Neither can any other company.”

This would apply, for example, to all files stored in the Azure platform and Office365.

### 3.3.2 The Fourth Amendment

In the United States, where most of the cloud storage providers are located, the personal information of individuals is protected by the Fourth Amendment:

“The right of the people to be secure in their persons, houses, papers, and effects, against unreasonable searches and seizures, shall not be violated, and no Warrants shall issue, but upon probable cause, supported by Oath or affirmation, and particularly describing the place to be searched, and the persons or things to be seized.”

Once personal documents are shared with others, they are not protected any more by the Fourth Amendment and can be accessed by the government without the need for a warrant or demonstrating probable cause. This is called the “third party doctrine”. The application of the Fourth

---

<sup>31</sup>[http://www.fincen.gov/statutes\\_regs/patriot/index.html](http://www.fincen.gov/statutes_regs/patriot/index.html)

<sup>32</sup><http://www.zdnet.com/blog/igeneration/case-study-how-the-usa-patriot-act-can-be-used-to-access-eu-data>

<sup>33</sup><http://www.zdnet.com/blog/igeneration/microsoft-admits-patriot-act-can-access-eu-based-cloud-data/11225>

Amendment to email or cloud computing has not yet been addressed by the Supreme Court. Without any legal guidelines, uploading files to a cloud storage provider can be considered sharing, the uploaded data then is not considered private anymore:

“However, when the object of a search — tangible or not — is voluntarily turned over to a third party, the Supreme Court has held that a person loses their reasonable expectation of privacy in that object.” [Cou09]

For users storing their files inside the United States (e.g. because they are using Dropbox), the only way to guarantee the privacy of the uploaded data is to encrypt it locally with a personal key before uploading it. The providers can not be trusted to keep the data confidential, since they can be ordered by law enforcement to cooperate with ongoing investigations. As an example, in 2007 the secure email provider Hushmail<sup>34</sup>

“...modified their product to capture the passwords of the three suspects, which it then used to decrypt the 12 CDs worth of email that it provided to US law enforcement agents.” [Sog09]

---

<sup>34</sup><http://www.hushmail.com/>

## 4. THE SECURITY ENGINEER'S VIEW

This section defines a minimal set of security requirements and appropriate measures to reach them that cloud storage services have to meet to be considered sufficiently secure for usage. These requirements will be applied to the cloud storage services and compared to the results obtained in the upcoming analysis.

The security requirements include the interaction with the web application via browser, the actual data storage and transmission as well as basic features of the cloud storage client applications and special features such as file sharing and publication. Last but not least, some requirements have been defined concerning the minimization of collected data by various processes revolving around interaction with cloud storage services.

We concentrate on security requirements that are observable to us. In general, there are some more security requirements that will not be addressed here, e.g. logging by the cloud storage service provider or security clearance of his employees.

### 4.1 Registration and Login

Before customers are able to use a cloud storage provider to synchronize or back up personal data, they have to complete a registration process. Cloud storage providers usually require the creation of a user account before any services can be used. On the one hand, it is in the interest of the service provider to establish a single point of contact through which all subsequent configuration, logging and — above all — accounting will take place. On the other hand, a customer who wishes to entrust personal data to the service provider wants to be certain that he communicates with the intended service and — above all — establishes a relationship of trust and contracts the service provider to perform its duties as pledged. During the registration process, the service provider and the new customer agree upon *credentials*<sup>↑</sup> – these must later be used to log in and use the service. If at any time an attacker is able to eavesdrop on the communication, he might obtain the credentials, compromise the account and gain access to uploaded data. Beyond that, if an attacker is able to manipulate the messages exchanged between customer and service provider, he might act as a proxy and defraud both of them. In order to prevent these attacks, all communication between service provider and customer must be secured in terms of authenticity, confidentiality and integrity. The de-facto standard to achieve these goals on the web is to use the Transport Layer Security (TLS) protocol [DR08]. Since service providers need to authenticate themselves against the client machine by presenting a certificate, customers can examine it and use it to verify that they are really communicating with the intended service provider. That way, they have a means to

detect impending phishing attacks, where attackers host a website which looks very similar to the intended service and try to get users to enter their credentials.

In case of a security breach, the easiest way to minimize potential data theft is to limit data collection to the bare minimum needed to operate the service – this approach is also called “data minimization”. The absence of valuable personal data might even make the service less attractive to financially-orientated attackers.

When customers register to services which are free of charge, apart from the email address only a unique key like a user name is needed to tie the customer to an account. However, storage services that need to be paid for necessitate the collection of the customer’s accounting data. To protect these, additional security measures should be enforced by the service provider – or optionally a third-party payment service could be brought in to handle the accounting entirely.

Even if an attacker is not able to directly glean any credentials, he can still attempt to guess them. The login systems of service providers are publicly accessible and can therefore be used to attempt a *brute-force attack*↑ or *dictionary attack*↑ on the credentials. In order to prevent these kinds of attacks, service providers should enforce sufficiently complex passwords — ideally 12 characters long and containing letters, numbers and special characters. But even then, guessing attacks may prove successful given enough time. To this end, service providers should implement measures to make these attacks infeasible such as time penalties or a temporary account lock down after a certain amount of incorrect login attempts within a given time frame.

Under ideal circumstances, a service provider would provide an authentication method which not only relies on the knowledge of credentials but rather demands possession of a token such as a smartcard. Alternatively, an access code could be sent to another physical device such as a mobile phone which has to be entered in addition to the credentials in order to login. Such authentication schemes — usually called two-factor authentication — combine something that is known to the user, like username and password with something he owns, like a mobile phone or smartcard. Overall, these schemes can significantly increase the security level of the login process.

Assuming that an authentication mechanism with a sufficiently high security level has been established by the service provider, additional measures should be implemented to protect standard processes during account management.

The email used by the customer should be verified during registration by sending an activation link used to complete the process. This prevents a possible incrimination where an attacker registers using an email address which does not belong to him. In appendix B we describe an attack based on missing verification.

Frequently, customers forget the credentials used to log into the service and need a way to create or receive new ones. If the system were implemented in such a way that new credentials were directly created as soon as the customer requested them, an attacker could abuse the password-reset process to effectively bar a customer’s access to the service — this is also called a *denial-of-service*

*attack*. To prevent this possibility, a link leading to a password-reset form or temporary credentials which have to be changed directly after logging in could be sent via email to the customer. Service providers should refrain from using questions about the user’s social background like “name of the user’s pet” or “user’s first car” as a means to allow login to the system. Such questions facilitate *social engineering*<sup>↑</sup> attacks on user accounts if an attacker has some knowledge about the user’s social background.

But sometimes it is not even necessary for an attacker to break into a system in order to glean information about a service provider’s customers. Feedback from the web application such as careless error messages can be exploited for information gathering. For example, if the registration or the login process informs a user that a chosen email address or user name is already taken by another account, an attacker can effectively enumerate valid email addresses and sell the information to spammers. Worse, he can directly gather valid user names and only needs to guess the password for a successful account hijacking attack. To guard against this attack, the system should never reveal more information to users than absolutely necessary.

## 4.2 Transport Security

Cloud storage providers usually provide client software which assists users in setting up their synchronization or backup schemes on the local devices. The actual transmission of all data with the remote storage servers is also handled by the client software. Similar to the registration and login process described in Section 4.1, attackers may be able to steal credentials, learn the contents of private data or even manipulate it if the communication between the client and the remote server is not sufficiently secured. Therefore, the server must authenticate itself to the client and all communication should be encrypted and its integrity ensured.

It is important to use appropriate cryptographic functions. All primitives, like symmetric and asymmetric encryption functions and hash functions should be up to date. This includes the algorithms as well as their parameters, like key lengths<sup>35</sup>. If keys are generated this should be done by a secure high-entropic key generator.

Algorithms and protocols should always be public, as stated by *Kerckhoff’s principle*<sup>↑</sup>. Keeping these things secret is always a risk, that does not increase security but decreases it dramatically.

Developing a cryptographic protocol is a very difficult task. In the past, even protocols designed by well-respected experts have failed. So it is in most cases a bad idea to invent a new algorithm for a well known problem, especially if a widely accepted solution is available.

---

<sup>35</sup>For a comprehensive overview covering multiple official recommendations see:

<http://www.keylength.com/en/>

The standard protocol TLS offers an established solution for transport security. There should be severe reasons for replacing it by something else for the same task.

### 4.3 Encryption

The main reason to use a cloud storage provider — for both individuals and companies — is to always have a backup of valuable data which is off-premises yet easily accessible. Arguably a reason for this is that personal data or sensitive company data is highly valuable in either sentimental or financial terms. Before cloud storage became popular, individuals and companies had personal backup strategies to protect against data loss, but these almost always relied on additional physical devices usually at the same location as the original data. Since the stored data was under the control of its owner, protection of the data itself was not always regarded as imperative. Nowadays, data is often entrusted to cloud storage providers on servers visible on the public Internet. Frequent discoveries of security vulnerabilities facilitate successful external attacks with ensuing data thefts. Additionally, there can be internal attacks from within the cloud storage provider itself. Therefore, the data itself should be protected in such a way that even in the event of a successful attack, the contents of the stored data remain confidential. To this end, all data needs to be stored on the remote servers in encrypted form. There are several cryptographically secure encryption schemes available which can be used freely. Cloud storage providers often offer a general encryption of all data stored on their servers using a *company key* which is known only to them. This may prevent data theft from external attackers, but does not protect against any attacks which include theft of the encryption key or internal attacks conducted by personnel who are able to gain access to these keys.

Therefore, all data should be encrypted on the client system before the data is transmitted into the cloud using a key unknown to the service provider. Stand-alone software may be used to encrypt all data on the client system, but this has drawbacks: The software has to be installed, administrated and operated on all client systems in addition to the client software of the cloud storage provider. The key used to encrypt the data needs to be distributed to all devices which are used to access the stored data. In the event that this key is lost, the data can never be decrypted again. As a precautionary measure, all keys used to encrypt data could be integrated into some kind of key escrow system to guard against data loss.

All keys that are used for encryption should be generated at random resp. pseudo random. This requirement ensures that two cryptograms of the same clear text are different.

Some products use a password to derive a cryptographic key. This has the disadvantage that just a small part of the key space can be reached by passwords. To produce a good key more entropy is required. A good example for this principle is the TrueCrypt disk encryption system, where key

derivation is based on a password and on *keyfiles*. A keyfile can be any kind of file, for example the user's favorite song as MP3. The keyfile is cryptographically mixed with the password to get a high entropic input for the key derivation function. We call this requirement a *high entropic password based key generation*.

An signing of data by the user enhances security because it enables to user to verify his data.

The statements on cryptography given in Section 4.2 are valid for encryption, too.

## 4.4 File Sharing

Sharing files appears in three different flavors (cf. Section 2.1.4):

- (1) Sharing files with other subscribers of the same service.
- (2) Sharing files with a closed group of non-subscribers.
- (3) Sharing files with everybody.

In any case the service should describe clearly which flavor of sharing is used. This is not a technical point but it is important because our investigations show that the difference between file publication and file sharing with a closed user group is very unclear for users. Hence, data that is dedicated to a closed group may be revealed to the public.

Sharing files with selected people, as in the first two cases, creates a closed user group and the sharing user has the role of an administrator within this group. General security requirements for these cases are:

- (1) The files that are being shared should only be accessible to the closed user group that was decided by the sharing user.
- (2) It should also be possible to revert sharing for each individual file.
- (3) A list of files currently being shared by the user could be accessed in the web interface or in the client application.
- (4) It should be possible to deal with different access rights and at any time the sharing user should be able to grant, edit or remove individual access rights.
- (5) If client-side encryption is used, sharing files should not weaken the security level. In particular, the cloud storage service provider should not be able to read shared files.
- (6) If client-side encryption is used, a disinvited user should be excluded by cryptographic means from the closed user group. In particular, this means that an encryption key that is known by the disinvited user can no longer be used for the encryption of new files.

Sharing files with selected non-subscribers is usually accomplished by providing a URL which is distributed to the intended group. Knowing this URL means having the right to access the



file<sup>36</sup>. Additional credentials (e.g. a password) may be used to enhance security. The security requirements are:

- (1) The URL should be obfuscated:
  - (a) The URL should not contain any information about the user, the file or the folder structure. Otherwise it might be possible for an attacker to guess filenames or to gain information about registered users.
  - (b) If no credentials are used, the URL should contain a randomly generated unique identifier. Taking the average number of shared documents into account, there should be enough different possible values so that it is infeasible for an attacker to successfully guess valid links. If the identifier size is too small or the identifier is simply incremented for each published document, an attacker may iterate over all possible links and thereby access all currently published files.
- (2) If the shared file is hosted on a web server, the cloud storage service provider should take care to exclude the file from being indexed by search engines.
- (3) Ideally, a cloud storage provider would provide an option to secure shared files with credentials chosen by the user. This would not only disable access to anyone outside of the intended user group, but also prevent the indexing of any published files.

Sharing files with everybody has the security requirement to hide informations about user names.

Note: If there is no client-side encryption with keys individual to the user the service knows which clients share files even though the clients do not use a file sharing feature. In some constellations this may disclose a connection between users that should be kept secret. This problem is independent of deduplication. If the service provides client-side encryption with individual keys generated at random the problem does not exist.

## 4.5 Deduplication

Data deduplication is employed by many storage providers since it enables them to save large amounts of storage space, thereby reducing costs. The different types of deduplication which are being used have been described in Section 2.3.1.

There are also some privacy issues that should be kept in mind as demonstrated by [HPSP10]. However, these privacy issues can only occur if the cloud storage provider uses both client-side and cross user deduplication. If this is the case, the following attacks may occur:

---

<sup>36</sup>This principle is also used by other services, e.g the time management tool doodle provides URLs like <http://www.doodle.com/5zi894pn1s8qe411> to invite persons.

- (1) An attacker who has an account at the cloud storage provider can use the deduplication feature to learn which files are already stored at the cloud storage provider: He transmits a file and observes what happens. If his client software does not upload the file, the attacker knows that this file already exists at the cloud storage provider. That is, he knows that at least one other user has the same file, but he does not know which one.
- (2) The previously described attack may also be used to find out information about a specific customer of the cloud storage provider: Assuming the attacker knows that a specific user *A* stores his medical examination results at the cloud storage provider, and these medical examinations are inscribed in a standard form with simple yes/no questions. The attacker could then create different versions of such results of medical examinations by using the standard form. In each version he inserts the user's name and he answers the questions with either 'yes' or 'no'. After that, he consecutively uploads these files to the cloud storage provider and observes what happens. If one of these files is not uploaded, the attacker knows that user *A* already stored this file, and thus the attacker knows the results of user *A*'s medical examination.

It is therefore mandatory for the provider to include a mechanism to protect users by preventing these attacks.

One solution to reduce the privacy risks concerning data deduplication is presented in [HPSP10]: It is based on the introduction of a random threshold which will be assigned to every stored file and which is kept secret by the storage provider. Deduplication will only be done if the number of uploads of a file exceeds this file's specific threshold. An attacker who wants to learn if a specific file has already been uploaded by another user, has to repeatedly upload this file until deduplication is performed. But if the storage provider performs deduplication, the attacker cannot tell if deduplication has been carried out because another user already uploaded the file before or because the attacker himself was the first who uploaded the file and now exceeded the threshold. This solution provides the advantages of data deduplication (although incurring additional expenses) to both the provider and the user while not exposing the user to any potential privacy threats.

One of the attacks described in [MSL<sup>+</sup>11] is using the deduplication feature and in principle enables an attacker to download files of other users. However, the attacker would need to know the hash values of the files he wants to download. Using a well-known hashing algorithm with a sufficiently large hash size, the probability that an attacker can guess valid hashes for (random or specific) files is negligible.

Ideally, the provider would only use client-side deduplication within a single account or, when using cross-account deduplication, would always upload any files added by the user even if they

are already on the server, thereby disabling any useful feedback to the potential attacker. There are currently no known privacy issues when only server-side deduplication is being used.

## 4.6 Multiple Devices

In the time of ubiquitous computing devices, a typical user has multiple devices to access his data depending on his current location. This might include the computer he has at home, his computer at work and his smartphone. Still, he wants to be able to access all his data in its most recent version from the device that is currently being used. Therefore, multiple different machines will have to be associated with a single account. The way a new location is added to a cloud storage account is of paramount importance when considering the security of said account. Multiple devices appear in the context of the synchronization feature, but also in the context of backup or storage where multiple, independent devices can be managed within one account without sharing data.

During the installation on a new device, the user will have to provide the credentials he created during the registration process to associate the new device with his account. After the initial login, the credentials are usually stored locally. This trade-off between usability and security allows the user to directly use cloud storage applications without having to enter his credentials every time. If the attacker is able to steal the credentials from the user (for example by intercepting user name and password over an unsecured connection), he could use those to associate his device with the account of the user. If this is not noticed, the attacker would not only be able to access the data currently stored by the user but would also be able to see all changes the user makes in the future. In the case of the Dropbox “config.db”-attack<sup>37</sup>, it was sufficient to copy a single file from the computer of the user in order to gain access to his account. The attack was hard to detect since there was no notification shown to the user when the new device was added by the attacker. To prevent this kind of attack, any devices that are added to a cloud storage account, including the first, should have to be activated by the user. This could be done by sending an activation email to the address that was used to register the account.

Even if the credentials are kept secure and are never compromised, there might be other ways for third parties to gain access to the user’s data. For example, when a smartphone is lost that was used to access files stored at a cloud provider, the person who finds the phone will be able to access these files using the cloud storage application on the phone. Therefore, the user should be able to remove specific devices from his account. This should be done by showing a list of devices currently associated with the account, either in the client application itself, in the web interface of the cloud provider, or, ideally, on both. To make the management of attached devices easier,

---

<sup>37</sup><http://dereknewton.com/2011/04/dropbox-authentication-static-host-ids/>, independently discovered by [MSL<sup>+</sup>11]

the user should be able to choose a name for the new device he wants to add to his account, e.g. “PC@home” or “private smartphone”.

## 4.7 Update Functionality

Running outdated software poses security risks since vulnerabilities that have already been fixed in newer versions will still be present in the old version and could be exploited by an attacker. Therefore, the client software could regularly check for updates. If a new version is found a notification should be shown to the user who then can decide whether he wants to update the program directly or at a later time. Alternatively the client software could automatically download and apply updates without any user interaction. There should also be a change log of the client application, either available on the website of the provider or shown to the user in addition to the update notification.

For non-corporate users the scheme described above works fine. However, automatic updating is a mixed blessing. From a corporate point of view it may be considered as harmful because an automatic update mechanism is an optimal way to gain access in order to perform an advanced persistent threat. An attacker that works under cover at the cloud storage provider can produce an update which performs some unwanted actions. After this he produces a further update to smudge the attack. So a company should consider to choose a conservative update policy. This problem is inherent to all products we analyzed in this study.

## 4.8 Server Location

The cloud storage provider should indicate where its servers are located, i.e., in which country the user’s data will be ultimately stored. Ideally, the storage provider would offer different storage locations from which the user can choose. The consequences of data storage location with regard to privacy and legal issues have been discussed in Section 3.

## 4.9 Classification of Security Requirements

In the preceding sections, several security requirements concerning cloud storage providers have been described. We now classify these requirements in *mandatory* and *additional* requirements. Security requirements that need to be fulfilled so users can be relatively sure that measures against the most common threats have been taken are mandatory. The additional requirements are not absolutely necessary, but they potentially increase the security level of the storage service and are

as such desirable. Table III gives a complete overview grouped by the security and privacy aspects described above.

Security / Privacy Aspect	Mandatory Requirements	Additional Requirements
<b>Registration and Login</b>	<ul style="list-style-type: none"> <li>– communication confidentiality and integrity</li> <li>– strong passwords</li> <li>– server authentication</li> <li>– account activation</li> <li>– password reset activation</li> <li>– protection against username and email enumeration</li> </ul>	<ul style="list-style-type: none"> <li>– multi-factor authentication</li> <li>– data collection minimization</li> </ul>
<b>Transport</b>	<ul style="list-style-type: none"> <li>– communication confidentiality and integrity</li> <li>– server authentication</li> <li>– suitable cryptography</li> </ul>	
<b>Encryption</b>	<ul style="list-style-type: none"> <li>– client-side encryption of data</li> <li>– client-side encryption of filenames</li> <li>– non-deterministic generation of encryption keys</li> <li>– suitable cryptography</li> </ul>	<ul style="list-style-type: none"> <li>– client-side signing of data</li> <li>– password based key generation with high entropy</li> </ul>
<b>File sharing</b>	<ul style="list-style-type: none"> <li>– clear description which flavor of sharing is used</li> <li>– obfuscated link<sup>b</sup></li> <li>– no indexing by external search engines<sup>b</sup></li> <li>– reversible sharing</li> <li>– disinvited users are excluded by cryptographic means<sup>a</sup></li> </ul>	<ul style="list-style-type: none"> <li>– configurable access rights</li> <li>– list of currently shared files</li> <li>– optional file access authentication<sup>b</sup></li> <li>– files shared with non-subscribers not readable by cloud service provider<sup>a</sup></li> </ul>
<b>Deduplication</b>	<ul style="list-style-type: none"> <li>– deduplication threshold<sup>c</sup> OR single account deduplication</li> </ul>	
<b>Multiple Devices</b>	<ul style="list-style-type: none"> <li>– list of registered devices</li> <li>– manual device activation</li> <li>– manual device deactivation</li> </ul>	<ul style="list-style-type: none"> <li>– choosable device identifiers</li> </ul>
<b>Updates</b>	<ul style="list-style-type: none"> <li>– integrated, regular update check</li> <li>– user-initiated update or silent update</li> </ul>	<ul style="list-style-type: none"> <li>– detailed change log</li> </ul>

*continued on next page*

*continued from last page*

Security / Privacy Aspect	Mandatory Requirements	Additional Requirements
Server Location	– storage location information	– selectable storage locations

<sup>a</sup>only relevant if the service provides client-side encryption

<sup>b</sup>not relevant for published files

<sup>c</sup>only relevant if the service uses cross-user client-side deduplication

Table III. Overview of mandatory and additional security requirements

## 4.10 Further Threats

We complete the section on security with some considerations which will not be included in our security requirements but which should nevertheless be kept in mind when data is moved to the cloud.

### 4.10.1 Time Related Aspects

We start with three topics that are relevant wrt time.

- (1) *Downtime.* If business heavily depends on the availability of a cloud storage service then downtime may be a problem. Even if the service promises some bonus on the monthly rate, this may be much less than the costs produced by the downtime. Hence, if access to data is really critical, then usage of more than one storage service is recommendable. Please note, that some services have a common backend (e.g. Amazon S3). So the redundant services should have different backends.
- (2) *Time to restore.* Imagine, a hard disk has crashed. How long does it take to restore a full disk? To have some concrete data, let us assume a disk capacity of 500 GByte and an Internet connection with a rate of 16,000 kBit per second, and let us assume, that the real data transfer rate is 1 MByte per second (not 2 MByte). The time to transfer 500 GByte is ( $d$  means day):

$$\frac{500 \text{ GByte}}{1 \text{ MByte/s}} = 500 * 1024 \text{ s} = \frac{500 * 1024 \text{ s}}{24 * 60 * 60 \text{ s/d}} = 5.9 \text{ d}$$

Thus, it takes almost six days for restoring a 500 GByte disk, a quite usual size these days. If this is too long, consider a service which supports alternative ways for restoring, e.g. by sending a portable hard disk. Sending a DVD resp. a set of DVDs is not a real alternative, because that means that you must insert more than 100 DVDs into your computer.

(3) *Migration time.* When a user wants to switch between two storage services, it would be beneficial if some kind of migration support is offered. If the user wants to migrate data from one provider to the next, usually he has to download all his files from the old provider. After that, the data needs to be uploaded to the new provider. Depending on the amount of stored data, this might take a very long time.

The user can also be forced to switch to another provider e.g. when the old provider goes out of business or is acquired by another company. For instance, this happened with Humyo which was acquired by Trend Micro in June 2010. The service is continued but does not accept new users anymore. Existing users are encouraged to try out SafeSync by Trend Micro. Even in this case (where both services belong to the same company) the user was not able to transfer his data without downloading it all, and then uploading it again<sup>38</sup>.

Currently, we cannot see, that this will change in near future, because it is not in the interest of any provider to support leaving customers and because encryption schemes, file sharing rules and many other details are proprietary. We do not expect any legal order which has the power to change this.

#### 4.10.2 Advanced Persistent Threats

After describing very concrete security requirements in the last sections, we are closing with a short discussion on Advanced Persistent Threats (APTs). This will not result in additional requirements, but we want to achieve that users of cloud storage services are aware of APTs.

The English Wikipedia provides a definition of an Advanced Persistent Threat<sup>39</sup>:

“Advanced persistent threat (APT) usually refers to a group, such as a foreign government, with both the capability and the intent to persistently and effectively target a specific entity. The term is commonly used to refer to cyber threats, in particular that of Internet-enabled espionage, but applies equally to other threats such as that of traditional espionage or attack.”

In June 2010 the Stuxnet worm was discovered. It was distributed initially via Windows and targeted Siemens industrial software, in particular equipment for uranium enrichment. It is broadly assumed that Stuxnet is an example for an APT. Stuxnet is extraordinary because of its complexity and the effort that was needed to program this malware.

Other kinds of malware is much longer in use<sup>40</sup> Most probably, there is a certain number of malware which infected their victims silently and which is undetected. As seen with Stuxnet,

---

<sup>38</sup>See also <https://www.humyo.com/pages/en/online-file-storage-questions>

<sup>39</sup>[http://en.wikipedia.org/wiki/Advanced\\_Persistent\\_Threat](http://en.wikipedia.org/wiki/Advanced_Persistent_Threat)

<sup>40</sup>a small collection: [http://en.wikipedia.org/wiki/Timeline\\_of\\_notable\\_computer\\_viruses\\_and\\_worms](http://en.wikipedia.org/wiki/Timeline_of_notable_computer_viruses_and_worms)



Malware has evolved from the purely destructive purpose and is created to fulfill a certain task, which might be sabotage, stealing of credit card numbers or espionage.

For an organization having enough money, enough time and a big pool of well educated people, cloud storage services may be a preferred target for placing a secret agent which works most of the time like any other employee but on activation he will do something harmful. For example, if data is stored unencrypted he may take a copy for his principal. Or he may slightly alter existing files according to his instructions.

One may think using a client-side encryption scheme, as offered by some of the analyzed services, may prevent from theft of data. We will cover this topic in the next section.

#### 4.10.3 A Note on Client-side Encryption

Some of the examined services provide a client-side encryption feature, that is the service does not know the key which is used for encryption. From our point of view, this is a very appreciated and required feature for a secure storage service. But is it also sufficient? Sufficient means, there is no need to trust the service provider because cryptography protects the data. Strictly speaking, the answer is no. In practice, a user should think about the amount of trust he is willing to place in his provider. The most important factor that requires trust, is the client software, which is distributed by the storage provider. A broken or manipulated client software is a risk. To see this point, consider the following scenarios:

- (1) *Key disclosure.* The client software receives encrypted data from the cloud storage provider. It uses the decryption key, either a symmetric or an asymmetric one, to obtain the clear text. A client software might send this key to the provider (or some other unauthorized party).
- (2) *Clear data.* Before files are sent to the cloud, the client software encrypts all of them. A client software may skip the encryption task to send clear data to the provider (or some other unauthorized party).
- (3) *Unconfigured directory.* Usually, the user configures one or more directories which are used by the client software for uploading to the cloud. A client software may take a file from somewhere else for uploading, in a worst case omitting encryption.
- (4) *Wrong public key.* Some services are using public keys for file sharing purposes. That means, if Bob wants to share a file with Charlie, he asks the service for Charlie's public key, encrypts a decryption key for Charlie, sends this cryptogram to the provider and finally, the provider sends the cryptogram to Charlie. In principle, this is a good idea. The problem is, that Bob can not verify the authenticity of Charlie's key, because there is no independent public key infrastructure. So in the worst case, he encrypts the decryption key for some other person, which consequently can decrypt files.

- (5) *Manipulated file content.* If an encryption is based on public key cryptography, public keys are known by some parties, including the provider. A server software may encrypt some contents using Bob's public key. When Bob downloads the file from the cloud, there is no automatism which allows him to detect the fraud. The only way to accomplish that, is to check data manually. In practice, nobody will do that. The problem is caused by the fact, that data is usually<sup>41</sup> not signed.
- (6) *Outdated version.* If signed data is used (and the signing key is really secure) it is not possible to create arbitrary contents, but a server software may send an outdated version of a file. Bob can not detect this case by any automatism.

We do not want to presume, that any cloud storage provider is evil. There are some other possibilities which lead to an undesired client software, for example, a bug in the software. Alternatively, consider an organizational bug, which leads to the distribution of an untested version of the client software, which is just in its beta phase.

Finally, to come back to advanced persistent threats, a secret agent, working at the provider may manipulate the client software. This is a promising way to inject malware in the customer's system. The automatic update function of the client software opens his system. The agent may produce the malware variant of the client software just for his victim. The malware then sends data to the agent (or some other unauthorized party), bypassing the encryption scheme, as mentioned above. Shortly after the agent has received the data he produces another variant of the client software which is a regular version and which destroys any traces of the attack. This is just a simple sketch in order to illustrate possible threats, of which there are much more than the one described here.

---

<sup>41</sup>exception: Wuala signs data.

# PART II: ANALYSIS OF CLOUD STORAGE SERVICES

## 5. METHODOLOGY FOR ANALYSIS

### 5.1 Selection of Products

The study does not aim at providing a comprehensive overview of existing cloud storage services, comparing and rating the different technical features, their ease-of-use and their prices. In fact, these approaches are regularly undertaken by print or online computer magazines<sup>42</sup>, knowledge bases<sup>43</sup> or dedicated websites<sup>44</sup>. Due to the constantly changing market, these periodically updated sources are better suited to keep users informed. Instead, a few cloud storage services were chosen for the following analysis. The services are:

CloudMe, CrashPlan, Dropbox, Mozy, TeamDrive, Ubuntu One, Wuala.

The services Dropbox and Mozy have been included since they are among the most popular cloud storage providers. They received broad media coverage and may well represent the gist of how cloud storage services are seen today: Very simple to set up, convenient to use and available for free or for a low price. These services have been designed as stand-alone cloud storage services and provide limited integration into popular operating systems.

On the other hand, IT companies developing operating systems now also offer cloud storage services such as Microsoft's SkyDrive, Canonical's Ubuntu One or iCloud from Apple. They have been designed to tightly integrate into their respective operating systems and directly provide a range of online services. It can be assumed that in the future these services will emancipate themselves from any single operating system and become very similar to stand-alone services. The analysis includes Ubuntu One, which has started as an additional service to the Ubuntu Linux operating system and is already moving to provide Windows integration as well.

The aforementioned services are all US-based or at least store their data in the US, which may cause some legal issues as described in section 3. As a counterbalance, three additional services have been chosen: CloudMe from Sweden and TeamDrive<sup>45</sup> from Germany which operate and store data exclusively within the EU. Wuala operates and stores all data in Germany, France and Switzerland<sup>46</sup>.

---

<sup>42</sup>Cloud Computing Magazine: <http://cloudcomputing.sys-con.com>

<sup>43</sup>[https://secure.wikimedia.org/wikipedia/en/wiki/Comparison\\_of\\_file\\_hosting\\_services](https://secure.wikimedia.org/wikipedia/en/wiki/Comparison_of_file_hosting_services)

<sup>44</sup><http://onlinebackupdeals.com/online-backup-comparison>

<sup>45</sup>for non-EU customers, TeamDrive stores data in the US

<sup>46</sup>By decision of the European commission, Switzerland ensures an adequate level of protection of personal data, see [http://ec.europa.eu/justice/policies/privacy/thirdcountries/index\\_en.htm](http://ec.europa.eu/justice/policies/privacy/thirdcountries/index_en.htm)

These services also offer enhanced functionality compared to the services mentioned above. CloudMe provides a whole web-based operating system within the browser complete with graphical desktop and applications for everyday needs, representing a first glimpse into a possible future of cloud computing where thin clients interact with applications running in the cloud.

TeamDrive, CrashPlan and Mozy are services offering a complete cloud-based backup solution sometimes superior to simple file sharing and synchronization services. TeamDrive and CrashPlan offer a dedicated, stand-alone server software which can be downloaded, installed and operated in private data centers. Wuala is the sole service that offers the full range of features as described in Section 2.1.

If customers are as yet hesitant to use the service provider's public cloud, using the dedicated server eases a mid-term migration to hybrid or public clouds.

## 5.2 Scope

The analysis of the individual cloud storage providers has been carried out in the following way:

The service's pricing model and different subscription plans as well as their costs and features are summarized. The technical capabilities and organizational details of the server and client software are also listed. The collected information stems from official services, pricing overviews and official documents that have been made available such as the *Terms of Service* and *Privacy Policy*.

All references and promises regarding the privacy and security guaranteed by the provider have been cross-checked by a technical analysis of the server's web application as well as the available client applications. While the analysis included a close look at the processes and their implementations, no professional penetration testing has been conducted. Usage of automated web vulnerability scanning on the infrastructure level has therefore been excluded.

Analysis of client software comprised the installation process, the subsequent configuration using the graphical interface or textual configuration files. Outside of the analysis' scope was a de-compilation of the binary executables and a security analysis based on these results.

In an effort to support a wide range of client devices, many cloud storage providers offer custom software for smartphones or tablets either directly or through official markets and stores for the respective mobile operating systems. However, the deduction of suitable security requirements in mobile scenarios and additional analysis of all client implementations are outside the scope of this study, but may be undertaken in the future. Availability of client software for mobile devices has been included to the best of our knowledge.

The communication between the client software and the server has in most cases been recorded — using suitable tools such as Wireshark<sup>47</sup> — and later analyzed to gain knowledge on how the security and privacy during data transmission is achieved. If a promising opportunity to test several attacks could be identified, a manipulation of transmitted messages has been undertaken using suitable tools such as WebScarab<sup>48</sup>.

All obtained results have been used to determine if the minimal and/or ideal security requirements as described in section 4 have been met. If security or privacy issues were detected during the analysis, the cloud storage provider has been notified in a responsible disclosure fashion and given the chance to fix the problem and/or provide an official statement.






Any open questions that remained after the analysis of a particular cloud storage provider have been compiled and sent through the official support channels.

### 5.3 Format

In the next sections, each analysis starts with a synopsis of features as described in Section 2.1 and supported operating systems, e.g.:

Copy	Backup	Sync.	Sharing							
	✓			✓	✓	✓			✓	

The icons used for operating systems in the right half have the following meanings:

				
Windows	Linux	Mac OS X	iOS	Android

If a cloud storage service can be used by a web browser or provides an application programming interface this is denoted by the following symbols:

	
Browser	API

The second part of the synopsis cares about security. It looks like:

<sup>47</sup><http://www.wireshark.org/>

<sup>48</sup>[https://www.owasp.org/index.php/Category:OWASP\\_WebScarab\\_Project](https://www.owasp.org/index.php/Category:OWASP_WebScarab_Project)

Reg.	Transport	Encryption	Sharing	Dedup.	Devices	Update	Location
+	+	±	%	++	±	++	--

The categories in the top line are taken from table III (p. 46). For each category we give a grade: “+” means good, that is, all mandatory requirements are met. “++” means very good, that is, all mandatory and at least one of the optional requirements are met. “±” means okay with some weaknesses, that is, most but not all of the mandatory requirements are met. “-” means bad, that is, at least one very important mandatory requirement is not met. “--” means very bad, that is, multiple important mandatory requirements are not met. “%” means the topic is not applicable. “Reg.” is short for “Registration” and “Dedup.” for “Deduplication”.






Following the synopsis, there will be a subsection “Availability” which describes which operating systems are supported and the pricing models.

After that the features of the service will be described.

The last part of every product section is “Security” where we examine the security features of the product, based on the requirements given in table III.

## 6. CLOUDME

### 6.1 Synopsis

Copy	Backup	Sync.	Sharing			X	iOS			
✓			✓	✓	✓	✓	✓	✓	✓	
Reg.	Transport	Encryption	Sharing	Dedup.	Devices	Update	Location			
--	--	--	-	%	-	+	+			

### 6.2 Availability

CloudMe<sup>49</sup> is operated by Xcerion<sup>50</sup>, which is located in Linköping, Sweden.

**Operating Systems** The CloudMe Web Desktop<sup>51</sup> is fully supported only when using Internet Explorer Version 7 or newer on Windows. Other browsers can be used to access the CloudMe service through the Simple WebUI (alpha)<sup>52</sup>, which is basically a file explorer with a very limited functionality. The CloudMe storage can also be accessed using the WebDAV protocol. Additionally, there is a tool called Easy Upload which can be used to monitor local folders and upload changed files regularly. This tool is available for Windows, Linux, and Mac OS. Android and iPhone applications are available. There is also a special version of the CloudMe Web Desktop which is designed for use with mobile devices (CloudMe Lite<sup>53</sup>).

**Client Software Version** CloudMe provides the users with different tools: Easy Upload (Version 1.09), Simple WebUI (alpha), CloudMe Lite (Version 1.0.5 beta) and Web Desktop (Version 3.38 Beta).

**Pricing** CloudMe follows the *freemium* business model also used by many other cloud storage providers. The basic service is free and provides 3 GB online storage space. The storage space can either be extended to 25 GB (CloudMe 25 GB, \$ 49.99 per year) or to 100 GB (CloudMe 100 GB, \$ 99.99 per year). The file size in the Free and CloudMe 25 GB versions is limited to 150 MB. The CloudMe 100 GB version has no file size limit.

<sup>49</sup><http://www.cloudme.com>

<sup>50</sup><http://www.xcerion.com>

<sup>51</sup><http://www.cloudme.com/en/supported/desktop>

<sup>52</sup><http://www.cloudme.com/webui>

<sup>53</sup><http://cloudme.com/m>



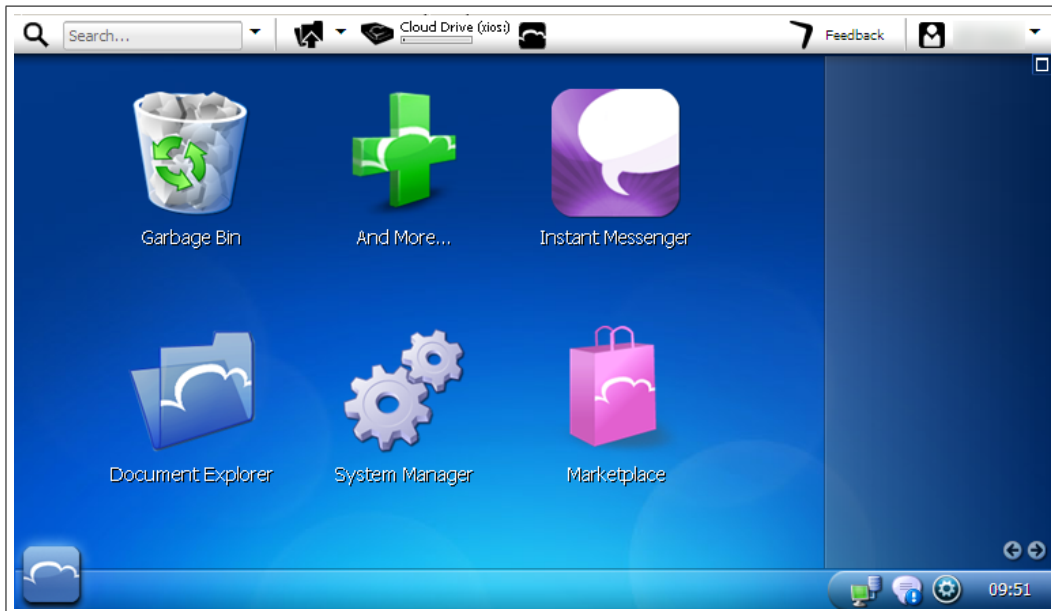


Figure 3. CloudMe Web Desktop

**Account termination** Xcerion reserves<sup>54</sup> the right to delete or deactivate the account, block any email or IP address or otherwise terminate access to or use of the CloudMe service without any notice and for any reason.

**Certifications** There is no information whether the CloudMe data center is certified.

## 6.3 Features

**Copy** With the EasyUpload tool, CloudMe can be configured to transfer files regularly to the cloud. The files that should be included can be selected with the EasyUpload tool, by default the directories `My pictures` and `My music` are selected for uploading (when using the Windows version). The user can select between a daily and weekly upload schedule, and uploading can also be started manually. Files can also be uploaded individually from the Simple WebUI, from the Web Desktop, or using the optional WebDAV access.

<sup>54</sup><http://www.cloudme.com/en/eula>

When using the Web Desktop or the Easy Upload tool to upload files, the file size limit is checked before the upload. When the file is too large an error message is displayed. When using the Simple WebUI, files that are larger than 150 MB are uploaded, but are not added to the online storage. After the upload is complete, depending on the browser that was used, a message Upload failed is displayed. When using WebDAV to access CloudMe, the upload of larger files is completed, but the files are not added to CloudMe. A HTTP 413 input filesize exceeded error is returned. If using the WebDAV-Tool suggested by CloudMe, files are added, but with a size of 0 Byte. Since no error is shown, users may be confused and think the upload was successful. This misinterpretation could lead to local file deletion and data loss.

The Web Desktop can be used to recover files, and individual files can be downloaded directly. Single or multiple directories can be downloaded after a zip archive containing the folder(s) has been created with the “compress” option in the Web Desktop. Recovery of files is also possible through the optional WebDAV access.

When files are deleted inside the CloudMe Web Desktop they are moved to a folder called Trash Can. From this location, the files can be restored. When files are deleted using WebDAV they cannot be restored.

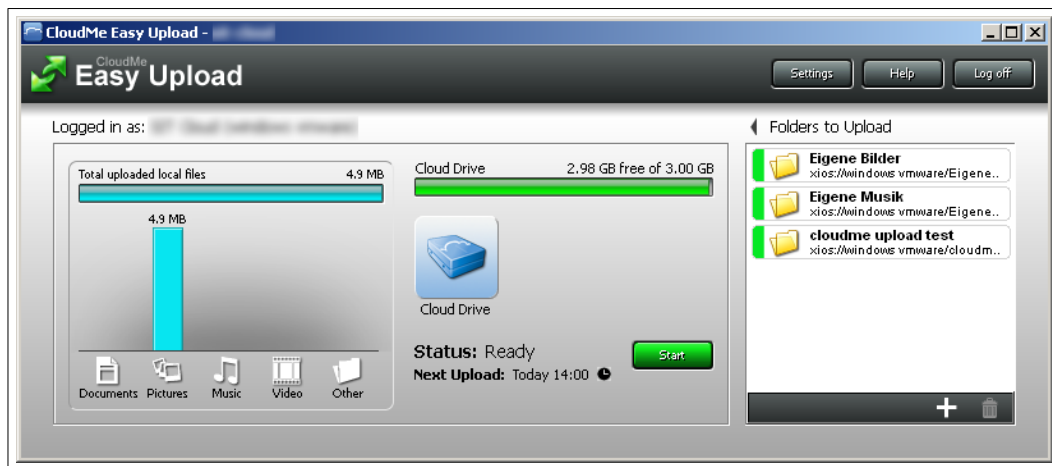


Figure 4. CloudMe Easy Upload Tool

**Backup** CloudMe does not offer a backup service as defined in Section 2.1.2.

**Synchronization** CloudMe does not offer synchronization.

**Sharing** CloudMe provides different ways to share data:

(1) *Sharing files with subscribers.*

Sharing of files with other subscribers is only possible when using the CloudMe Web Desktop. To share a folder or file with another user the owner of the data right-clicks only the item he wants to share and selects `Sharing...`. From there, existing friends of the owner can be select to share the data with. The share will automatically be available to the invitees under `Friends` → *friend* → *FileOrDirName*, where *friend* is the invited user. Different access rights (Read, write, delete, edit ACL) can be configured for each file or folder by right-clicking again selecting `Properties...` → `Security`. The invitee will also receive an email notification telling him that another user has shared files with him.

(2) *Sharing files with non-subscribers.* CloudMe provides two variants:

(a) *Sharing with non-subscribers, protected by a password.* This is done by right-clicking on a folder or a file in the WebOS Desktop and selecting `Share` → `Share...`. After that step a new window appears and the user can choose to make the new share hidden and/or password protected. The newly generated link points to a URL like `http://my.cloudme.com/username/webshare/FileOrDirName`. The password can be as short as one character only. Example: If user `john` is sharing a folder named `pictures` the link would look like `http://my.cloudme.com/john/webshare/pictures`.

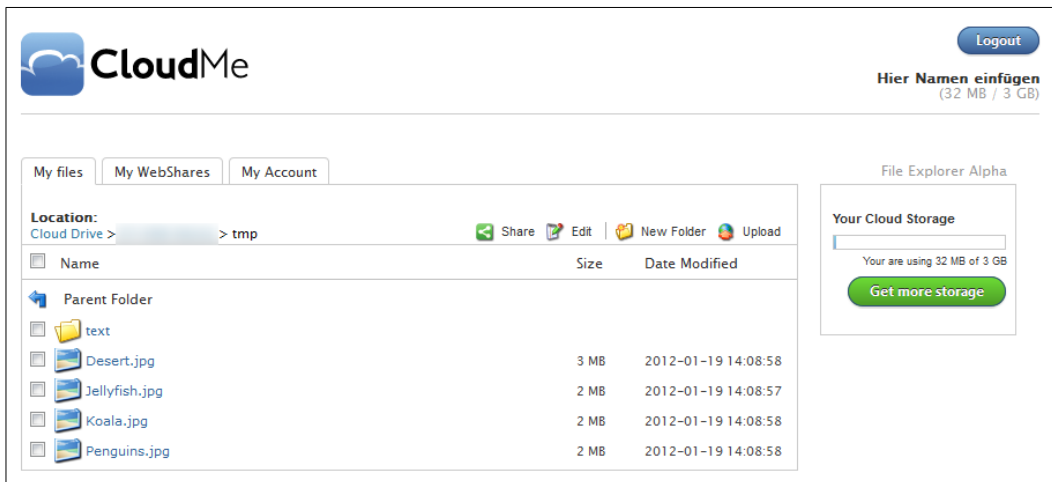


Figure 5. CloudMe Simple WebUI

- (b) *Sharing files with non-subscribers, protected by obfuscated URLs.* The user has to click on the file and select Properties... → Web Links → Create new. Files published like this will be available at URLs like `http://os.cloudme.com/v1/links/userid/fileid` where *userid* is a 11-digit numerical value which seems to be incremented for each user. *fileid* is a 5 digit numerical value which seems to be incremented for each file shared across the system.
- (3) *Sharing files with everybody.* The user copies files or folders he wants to publish in the folder `Public`. Files inside this folder are automatically published. Directory listing is enabled, so every not hidden share is shown to the public at the URL `http://my.cloudme.com/username/webshare`.

## 6.4 Security

### Registration and Login

For the registration with CloudMe, the user has to provide country, username, password, email, first and last name. The password has to be at least six characters long. This restriction is not displayed on the page and can only be found out using trial and error: if the password is too short, the password field will be highlighted in red and the registration can not be completed. The user is not given any additional feedback regarding the failed registration, i.e. the required length of the password. When entering the desired username on the registration page, the availability of the user name is checked automatically after every character. This allows the gathering of currently registered usernames. The email does not have to be unique, multiple accounts can be registered using the same email address. When completing the registration, username and password are transmitted in plain text over HTTP. The account is active immediately after registration, the email used to sign up for CloudMe is not validated. Appendix B (p. 133) describes an attack based on this weakness. CloudMe has been informed.

If a user does not remember the account password, a new password can be set on the CloudMe site. The user has to enter the username or email address for which he wants to start the password reset process. Then an email is sent to the user which contains a link to an individual password reset page. From this page, the password is transmitted in plain text over HTTP.

CloudMe enables information gathering regarding already registered usernames on multiple occasions. For example, the availability of a username entered during registration is checked after the input of every character, the result is shown to the user. The required password complexity of six arbitrary characters is not ideal. Also, this requirement is not shown to the user as was described above.

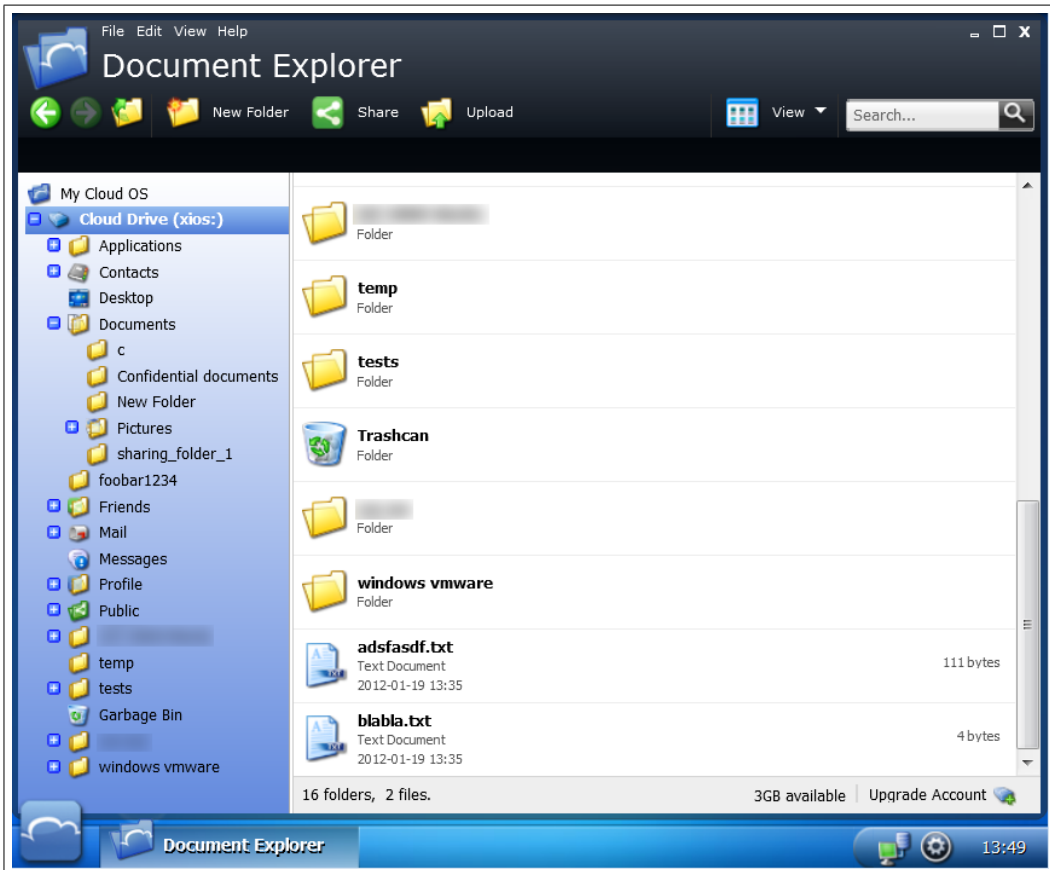


Figure 6. CloudMe File Explorer

During the registration process, both the username and the password are transmitted in plain text over HTTP and therefore could be intercepted by attackers.

## Transport Security

CloudMe does not encrypt the data transferred between the server and the client.

## Encryption

CloudMe does not encrypt the files that are stored on the server. Since the communication between the client and the server is also not encrypted, attackers are able to intercept every file a user uploads to the service.

## Sharing

- (1) *Sharing files with subscribers.* The sharing of files with subscribers can not be checked since it doesn't work correctly at the moment. Folders can be shared with other users of the CloudMe service, however the invited users can't
- (2) *Sharing files with non-subscribers.* The two variants are evaluated as follows:
  - (a) *Sharing files with non-subscribers, password protected.* A URL to access shared data (e.g. `http://my.cloudme.com/username/webshare/FileOrDirName`) contains username and filename resp. directory, hence does not meet our requirements. The required password length of one character is not enough to guard against any attacks .
  - (b) *Sharing files with non-subscribers, protected by obfuscated URLs.* A URL to access shared data (e.g. `http://os.cloudme.com/v1/links/userid/fileid`) contains the userid, hence does not meet our requirements. Additionally, since the fileids are simply incremented, the following attack is possible: After Alice shared one file this way with Eve, Eve can check if Alice has shared other files in the past by using the link up to the userid and iterating over all possible fileid values.
- (3) *Sharing files with everybody.* The published URL (e.g. `http://my.cloudme.com/username/webshare`) contains the username, hence does not meet our requirements.

Additionally, we have found serious vulnerabilities in the WebOS Desktop which have been communicated to CloudMe and have been fixed subsequently, see appendix A (p. 132). CloudMe has been informed.

## Deduplication

CloudMe does not use deduplication.

## Multiple Devices

The CloudWeb Web Desktop can be accessed from arbitrary machines. The EasyUpload tool can be used on multiple computers to upload files to CloudMe. During the installation the computer has to be assigned a unique name, which will be used to create a folder in CloudMe. Files chosen for backup during the installation will be uploaded to this folder, for files that are added later the storage location on the server can be selected. There is no central overview of installations.

## Update Function








The EasyUpload tool, which is the only client application used by CloudMe, is by default configured to automatically check for application updates.

## Server Location

According to the information on the CloudMe website, the CloudMe data center is located in Sweden, which has been verified by our analysis.

## 7. CRASHPLAN

### 7.1 Synopsis

Copy	Backup	Sync.	Sharing							
	✓			✓	✓	✓			✓	
Reg.	Transport	Encryption	Sharing	Dedup.	Devices	Update	Location			
+	±	+	%	+	±	++	++			

### 7.2 Availability

CrashPlan<sup>55</sup> is operated by Code 42 Software<sup>56</sup>, which is located in Minneapolis, USA.

**Operating Systems** Both the client application (see Figure 7) and the server application are available for Windows, Mac OS X and Linux. A web interface is also available, and can be used to upgrade accounts, to manage the information associated with the account and to configure individual installations of the CrashPlan software. Friends and computers added to the account can also be viewed online, and friends can be removed online. When using the CrashPlan server, files can be restored through the web interface.

**Client Software Version** The CrashPlan client software used during our tests was version 3.0.3. At the time of writing, this is the latest version.

**Pricing** CrashPlan offers a free service called *CrashPlan* which can be used to backup unlimited files locally, on other computers belonging to the user, or on the computers belonging to other users (“friends”) who are also using CrashPlan. The free version does not include any online storage space.

CrashPlan+ includes 10 GB online storage<sup>57</sup> for a single computer (\$ 1.50 per month) and can be upgraded to unlimited storage space for a single computer (CrashPlan+ Unlimited, \$ 3.00 per month) and to unlimited storage space for up to ten computers (CrashPlan+ Family Unlimited, \$ 6.00 per month). Apart from the online storage space, CrashPlan+ offers continuous backup, upgraded security (448-bit blowfish data encryption), web restore function and backup sets in addition to the features of the free version.

<sup>55</sup><http://www.crashplan.com>

<sup>56</sup><http://www.code42.com/>

<sup>57</sup>Online storage is referred to as CrashPlan Central in the CrashPlan client



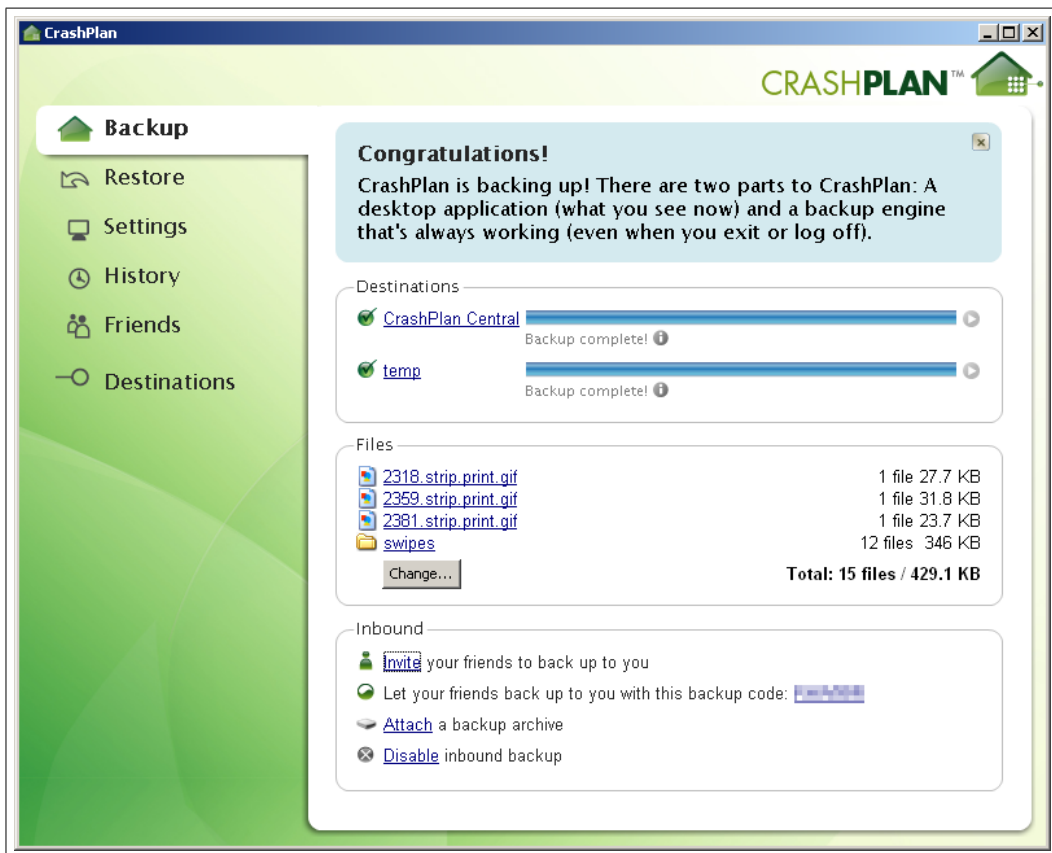


Figure 7. CrashPlan Interface

CrashPlanPRO is designed for business use and includes an online management interface for administrators which can be used to manage users. It supports up to 200 computers and individual logins. An unlimited plan is available for \$ 7.49 per month per user.

CrashPlanPROe is designed for enterprise use and offers a server application that can be used to host a CrashPlan storage server locally, for example to store files inside the company network. Additional functionalities included are custom installers, LDAP support and a REST API. Five client licenses are available for \$ 349.95 (including one year of support and upgrades, which costs \$ 62.00 for five users after the first year).

In addition to the features mentioned above, CrashPlan also offers both restore and upload via physical hard drive (1 TB size). For upload, the user will be sent an empty hard drive which he

can fill with data he wants to have backed up and then send the drive back to CrashPlan. For restore, the user will be sent a hard drive containing his backup and can restore his files locally (\$ 124.95)<sup>58</sup>.

**Account Termination** Code 42 reserves<sup>59</sup> the right to terminate the ability to continue to use Code 42 products. In the case of termination their products cease to function, which results in the user not being able to access any encrypted data stored using the CrashPlan software.

**Certifications** There is no information as to whether the CrashPlan data center is certified.

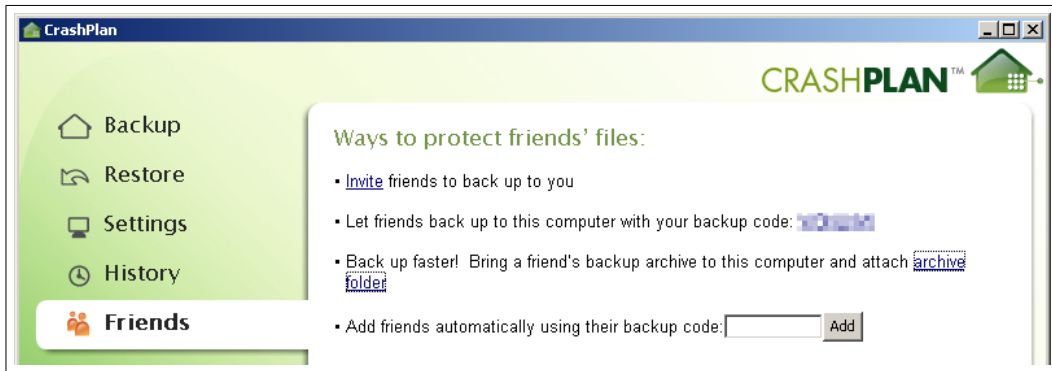


Figure 8. The friend feature in the CrashPlan client

## 7.3 Features

**Copy** CrashPlan does not support a copy feature as defined in Section 2.1.1.

**Backup** The basic CrashPlan service can be used to back up files locally (e.g. on a second harddrive), on other computers belonging to the user which also run CrashPlan or on the computers belonging to other users which also use CrashPlan. For the last option, the users have to be added as “friends”. This can be done by sending them an invitation email through the CrashPlan client or, if they are already using CrashPlan, by using the friend codes provided with every CrashPlan installation (see Figure 8). The relationship between friends can be one-way (storing files at friends / allowing friends to store files) or two-way. The backup process (either scheduled in the basic

<sup>58</sup>This feature is currently only available inside the US

<sup>59</sup><http://support.crashplan.com/doku.php/eula>

version or continuously in the advanced versions) copies all changed files to all configured backup locations. The files are encrypted locally before they are transmitted to any backup location.

Recovery can be done from every location (see Figure 9, here). The user can either restore the previous version or can select the date and time of the version he would like to restore. The destination folder for the restoration and whether existing files should be overwritten or renamed can also be configured.

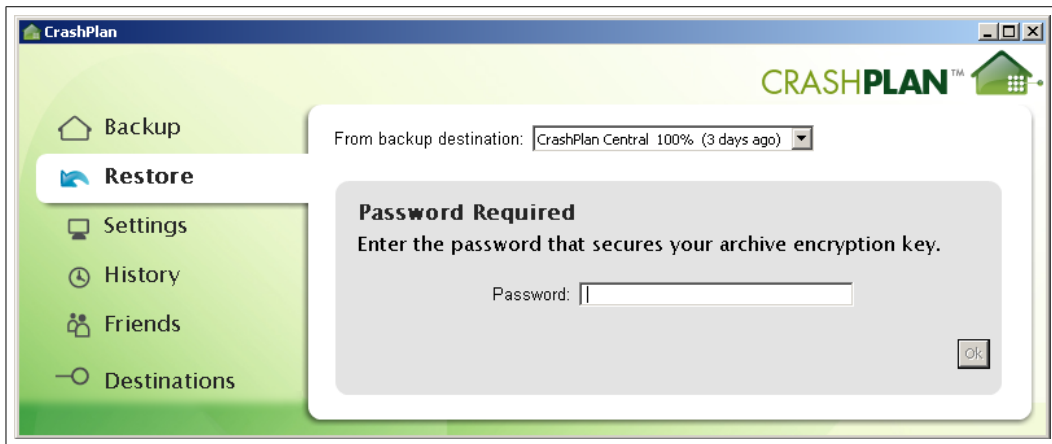


Figure 9. File restoration from the CrashPlan Interface

**Synchronization** CrashPlan does not support synchronization. Multiple computers can be attached to an account but they will back up their own data and even if two computers have the same copy of a file, the file will be uploaded twice for that single account.

**Sharing** CrashPlan does not support sharing of files.

## 7.4 Security

### Registration and Login

The CrashPlan account is created during the installation of the software. The user is required to enter first and last name, an email address and a password. The password has to be at least six characters long, there are no other restrictions. The strength of the password is displayed next to the field where the password is entered, with four different values:

(1) Very weak: shorter than six characters (not accepted)

- (2) Weak: Any password at least six characters long that only includes two different characters
- (3) Strong: Any password at least six characters long that includes more than two distinct characters
- (4) Very strong: Any password at least six characters long that includes letters, numbers and special characters (at least one of each).

Unless the password chosen by the user is already “Very strong”, CrashPlan displays hints on how to make the selected password stronger.

The account is active immediately after registration and does not have to be activated by the user. A “Welcome to CrashPlan” email is sent to the address used during the registration process.

CrashPlan enables information gathering regarding already registered email addresses and does not verify the email used during the registration. Because CrashPlan encrypts all files at the client with a key chosen during the installation, we do not see any attack based on the missing activation. Therefore, we did not downgrade CrashPlan because of this. The hints given to the user when he chooses the password are helpful, although we would not use the term *Very strong* to describe a password that has to include numbers, letters and special characters while being only six characters long.

## Transport Security

CrashPlan does not use SSL/TLS to secure the communication between a client and the CrashPlan server, instead a self-made, unpublished protocol is used. This is a violation of *Kerckhoff’s principle*<sup>↑</sup>. The communication between the client and other backup destinations is not secured by SSL/TLS. This is a disadvantage if these destinations are outside of an intranet.

We devaluate for the non-published protocol and for the missing secure channel to other backup destinations.

## Encryption

The key used to encrypt the files is chosen at random during the installation of the software and will be referred to as *data key*. CrashPlan provides multiple options to encrypt files, which are explained in high detail on the CrashPlan website<sup>60</sup>:

- (1) *Securing the data encryption key with the account password*. This is the default setting. The account password, which is known to CrashPlan, is used to secure the data key. The encrypted data key is stored on the CrashPlan server and will be transferred to additional CrashPlan installations automatically. This also enables web restore function.

---

<sup>60</sup>[http://support.crashplan.com/doku.php/articles/encryption\\_key](http://support.crashplan.com/doku.php/articles/encryption_key)

- (2) *Securing data encryption key with a private password.* In this setting, the user chooses a private password which is used to secure the data key. The private password is not known to CrashPlan. The encrypted data key is stored on the CrashPlan server (and on other destinations for guest restore) and will be transferred to additional CrashPlan installations automatically. Using this option, a key to encrypt the data encryption key is derived from a password but without additional high entropy (cf. 4.3, p. 39).
- (3) *Using an exclusively local stored data encryption key.* With this setting, the user chooses a private data key, which is only stored locally. The data key is never stored at any destination and has to be managed by the user himself. It is also possible to choose a separate private data key for every CrashPlan installation.

With the default option, it is possible for CrashPlan to decrypt and access the data stored on their servers, since both the data key and the password used to secure the data key are known to CrashPlan. With the second option, CrashPlan can't access the encrypted data unless the user is using the web restore function, where he has to enter his private password which is then used to unlock the data key. This option does not implement a high entropic password based key generation<sup>61</sup> (cf. Section 4.3, p. 39), so we give a "+" for encryption instead of "++". Using the third option, the private data key has to be entered when using the web restore function. The user is responsible to store this key in a secure and safe way.

The security level can only be upgraded, it is not possible to switch back to a lower level. Changing the default security level to *private password* requires a double opt-in by the user, he has to confirm that he understands that the security can not be downgraded later, and that he understands that there is no way to restore his data if he loses his password. The highest level additionally requires a complete backup to be done, since the files are being encrypted with a new data key. Figure 10 shows the notice displayed to the user when switching to the private data key option.

The different encryption options, the included encryption of filenames and especially the very detailed explanation of the benefits and drawbacks to these security features are exemplary. However, there could be additional support to the user when he chooses the private data key option and a better password based key generation.

## Sharing

CrashPlan does not support file sharing or file publication. When using the friend-feature to store files on the computers of other users, these files are stored encrypted and are not accessible to the other users.

---

<sup>61</sup>Of course, neither the first options does, but the the requirement does not make sense for it.

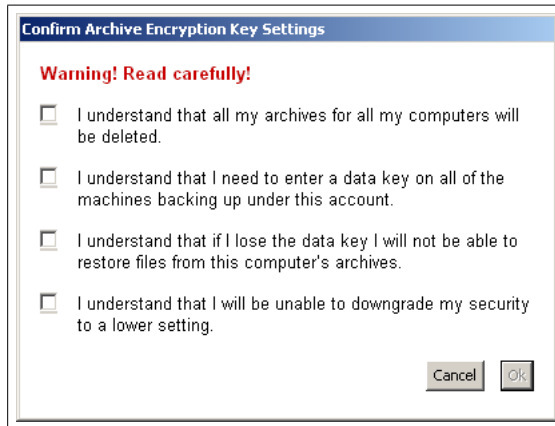


Figure 10. User notification when changing to private data key

## Deduplication

CrashPlan uses single-account deduplication, which has no privacy issues. Deduplication can also be disabled in the paid versions.

## Multiple Devices

For CrashPlan accounts that allow access from multiple locations, the different installations associated with the account can be managed in the web interface (see Figure 11). Here it is possible to configure the installations almost exactly like in the client, including backup schedule, general settings and alerts and notifications. The changed settings are synchronized with the corresponding installation. It is not possible to remove individual installations.

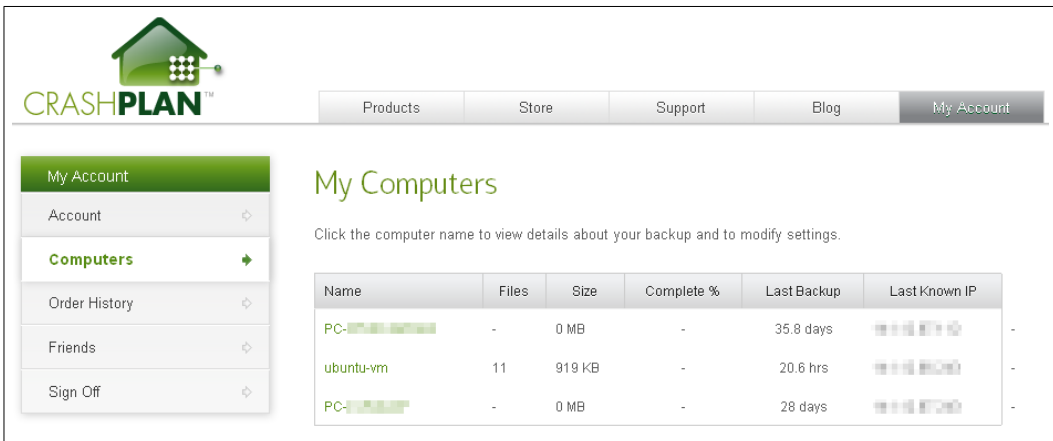
## Update Function

CrashPlan automatically updates to the latest version of the client software.

## Server Location

According to the CrashPlan website, the data center is located in Minneapolis, which has been confirmed by our analysis. There is also a data center overview<sup>62</sup> available on the website which gives extensive information about the data center including security and redundancy mechanisms.

<sup>62</sup><http://www.crashplan.com/consumer/features-datacenter.html>



**CRASHPLAN**™

Products Store Support Blog My Account

My Account

- Account
- Computers**
- Order History
- Friends
- Sign Off

## My Computers






Click the computer name to view details about your backup and to modify settings.

Name	Files	Size	Complete %	Last Backup	Last Known IP
PC-██████████	-	0 MB	-	35.8 days	██████████
ubuntu-vm	11	919 KB	-	20.6 hrs	██████████
PC-██████████	-	0 MB	-	28 days	██████████

Figure 11. Overview of individual computers associated with one account

## 8. DROPBOX

### 8.1 Synopsis

Copy	Backup	Sync.	Sharing			X	iOS			
✓		✓	✓	✓	✓	✓	✓	✓	✓	✓
Reg.	Transport	Encryption	Sharing	Dedup.	Devices	Update	Location			
-	+	-	±	+	±	++	+			

### 8.2 Availability

Dropbox<sup>63</sup> is operated by Dropbox Inc., which is located in San Francisco, US.

**Operating Systems** The client application runs on Windows, Mac OS X and Linux. A web interface is provided which may be used for account management as well as data access. Android, Blackberry, iPhone and iPad applications are also available.

**Client Software Version** The client software that has been used in our tests was version 1.1.35. The latest version is 1.2.49.

**Pricing** Dropbox offers a free service providing up to 2 GB storage space (Dropbox Basic). Additionally, two premium services providing up to 50 GB storage space (Pro 50, \$ 9.99 per month) and up to 100 GB storage space (Pro 100, \$ 19.99 per month) are offered. Dropbox for Teams is a special offer for companies and organizations, and comes with additional administration features (5 users and 350 GB shared storage space: \$ 795.00 per year).

**Certifications** Dropbox uses Amazon Web Services (AWS) for storage and transfer which is SAS 70 Type II certified.

### 8.3 Features

**Copy** During the installation of the client application, the user has to choose a local Dropbox folder. All files in this folder and all subdirectories will be uploaded. The process starts immediately after installation and runs permanently in the background.

<sup>63</sup><http://www.dropbox.com>



Files can be added to the backup by adding them to the Dropbox folder. It is not possible to include files or folders outside of the Dropbox folder. Files can also be uploaded through the web interface. Files can be restored from the web interface.

Dropbox keeps records of previous versions of a file and can restore to any version. This restoration can only be done in the web interface and for single files only. The records are kept for 30 days. Unlimited undo support is available for additional charges.

**Backup** Dropbox does not support a backup feature as defined in Section 2.1.2 by default.

**Synchronization** The user can install the client application on multiple computers and the data will automatically be synchronized on all these computers. Dropbox recognizes conflicts during synchronization. In case of a conflict, a new copy of the file is created and stored in the user's Dropbox folder. Conflicting files are renamed including the date of the conflict and the device from which the conflicting version was uploaded. The user has to compare the files by himself and resolve the conflicts manually.



Figure 12. Conflicting files

**Sharing** Files can be shared with subscribers of Dropbox. Further, files can be copied in a public folder in order to obtain a URL that allows access for non-subscribers of Dropbox. The service is unclear, whether this creates a closed user group or is meant as file publication. The statement “It is possible, however unlikely, that someone could guess your link if they knew the file name.”<sup>64</sup> makes thinking that a closed user group is intended. On the other side, Dropbox says “Everything in your Public folder is, by definition, accessible to anyone.”<sup>65</sup> which may be seen as an indicator for file publication. This ambiguity can cause a problem.

(1) *Sharing files with subscribers.* This is done by inviting the users by entering their user name or email address. If there is no account registered with an invited email address, a registration invitation is sent to this email. It is not possible to assign individual permissions for the invited users, but invited users are not able to permanently delete files or remove versions of individual files. The `sharing` tab inside the web interface provides an overview about folders that are currently being shared by the user or folders to which the user has been invited.

<sup>64</sup><https://www.dropbox.com/help/179>

<sup>65</sup><https://www.dropbox.com/help/27>

It is possible to leave the folder while keeping a local copy of the files. The inviting user can remove invited users from the folder and can choose whether the removed users should be allowed to keep local copies of the files inside his Dropbox folder.

- (2) *Sharing files with non-subscribers / everybody.* Files are shared by copying them to the specific `Public` folder. This folder is mapped to a URL like `http://dl.dropbox.com/u/n`, with  $n$  being a 7-8 digit number. Links to files include their original filename (e.g. `http://dl.dropbox.com/u/47110815/example.jpg`).

## 8.4 Security

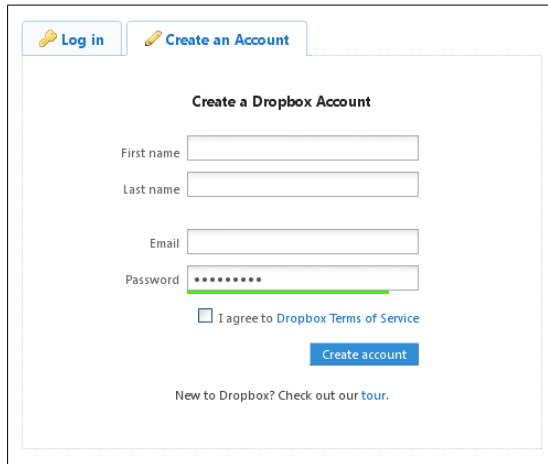
### Registration and Login

Both the registration and the login process use secure communication channels (TLS). During the registration, which can either be done on the website or during the client installation, the user has to enter a first and a last name (both arbitrary strings), an email address, and a password. The email address will be used to login to the service and there can only be one account associated with this email address. If an already registered email-address is used during the registration, the message “This email address is already taken.” is shown to the user. Dropbox accepts weak passwords; the only restriction is a minimal password length of six characters, the email address must not be used as password.

When registering on the Dropbox web site, Dropbox gives a hint about the quality of the chosen password in the form of a colored bar (see Figure 13). To get an indication for a strong password, the user has to choose a password consisting of characters from different categories (lowercases, uppercases, digits, and special characters). However, selecting more than three different characters from the same category does not continue to increase the indicated password strength. The user is not prompted to repeat the password in order to prevent typos. The registration process during the client installation slightly differs: The client application has no password strength indicator and the user has to repeat the password. Dropbox does not send any activation emails after registration. This enables an incrimination attack. The user may use the new account immediately after completing the registration form.

During the login process, the user has to enter his email address together with his password. In case of an incorrect login attempt, Dropbox only informs the user, that one of the two is incorrect but not which one (see Figure 14).

When logging in to the client application for the first time, the user is prompted to enter his email address and the password. After the user is authenticated, a token is sent by the server and stored on the client which is used to authenticate the user from there on. Note that up to client version



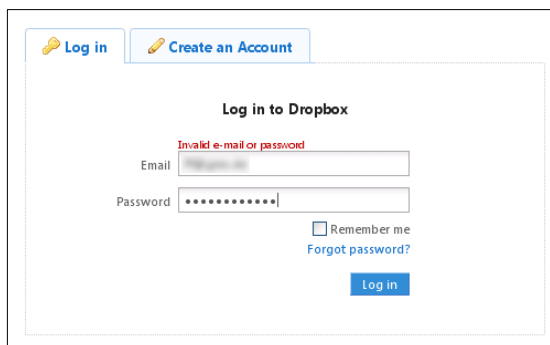
The screenshot shows the 'Create a Dropbox Account' form. At the top, there are two buttons: 'Log in' (with a key icon) and 'Create an Account' (with a pencil icon). The form title is 'Create a Dropbox Account'. It contains four input fields: 'First name', 'Last name', 'Email', and 'Password'. The 'Password' field has a green progress bar below it, indicating password strength. Below the password field is a checkbox labeled 'I agree to Dropbox Terms of Service'. A blue 'Create account' button is positioned to the right of the checkbox. At the bottom, there is a link: 'New to Dropbox? Check out our tour.'

Figure 13. Dropbox: Hint about password quality

1.1.35, an attacker who succeeds in copying a victim's configuration file to his own machine, will have access to the victim's Dropbox account<sup>66</sup>.

Dropbox repels brute force password attacks: Dropbox temporarily locks an account after too many failed logins in a given time frame.

If the user forgets his password, Dropbox sends an email to the email address registered with the user's account. This email contains a link to a secure website for entering a new password. The account will not be changed unless the entire password reset process is completed by the user.



The screenshot shows the 'Log in to Dropbox' form. At the top, there are two buttons: 'Log in' (with a key icon) and 'Create an Account' (with a pencil icon). The form title is 'Log in to Dropbox'. It contains two input fields: 'Email' and 'Password'. The 'Email' field has a red error message above it: 'Invalid e-mail or password'. Below the password field is a checkbox labeled 'Remember me' and a link 'Forgot password?'. A blue 'Log in' button is positioned to the right of the 'Remember me' checkbox.

Figure 14. Dropbox: Failed login attempt

<sup>66</sup>This is commonly called the "config.db attack" and has been fixed now

Dropbox' registration process has some minor weaknesses and therefore does not completely meet the requirements defined in Section 4: Dropbox accepts weak passwords and the email address used to sign up for the service is not verified. In appendix B (p. 133) we have described an attack based on this weakness. Dropbox has been informed.

Dropbox' measures to prevent information gathering could be improved; especially during the registration process, gathering of email addresses of already registered users is possible.

A bug in the client application which will enable an attacker to get access to a victim's Dropbox account is fixed in the current version.

## Transport Security

Dropbox uses TLS to encrypt the communication between the client application and the server. The communication between the browser and the web interface is encrypted by using HTTPS.

## Encryption

Dropbox uses AES-256 to encrypt data stored on its servers. The data will not be encrypted at the client; instead Dropbox encrypts the data after the upload on the server-side using its own encryption key.

While the encryption of data in transit meets the requirements, Dropbox has not optimally implemented the encryption of the stored data. Since Dropbox itself encrypts the data on the server-side, users cannot be sure by cryptographic means that all stored data is highly confidential.

## Sharing

Dropbox has some problems when sharing files with non-subscribers / everybody.

- (1) *Sharing files with subscribers.* This meets our requirements.
- (2) *Sharing files with non-subscribers / everybody.* The shared URLs look like `http://dl.dropbox.com/u/n/f`, with *n* being a 7-8 digit number, and *f* the filename, as described above. URL analysis of multiple files being published revealed that the numbers seem to be incremented but the lack of file name obfuscation enables easy access by anyone. Using a simple script which iterated through possible URL combinations we were able to search for the existence of specific files inside the `Public` folder. This is a contradiction to the statement that it is unlikely to guess filenames (Section 8.3, p. 73). Additionally, the shared files are not excluded from search machine indexing<sup>67</sup>. We downgrade Dropbox wrt sharing because of the unclear definition of sharing.

---

<sup>67</sup>Using Google as an example: `http://www.google.com/search?as_sitesearch=dl.dropbox.com`

## Deduplication

Currently, Dropbox only uses single user deduplication which has no privacy issues. The switch to single user deduplication was made when the program Dropship<sup>68</sup> became available, which enabled users to share large files via Dropbox simply by exchanging small hash values. The author of Dropship reverse-engineered the Dropbox deduplication protocol and used this information to create the program. Dropbox plans to enable this function again, but so far has not given any specific time line<sup>69</sup>.

The screenshot shows the 'My Computers' section of the Dropbox web interface. At the top, there are navigation tabs for 'Account Info', 'Account Settings', 'My Computers', 'Referral Status', and 'My Apps'. Below the tabs, the title 'My Computers' is displayed, followed by the text 'These are the computers and devices currently linked to your Dropbox.' A table lists the following devices:

Computer name	Last Activity	Actions
MacBook	1/12/2012 9:38 AM	Rename  Unlink
Mac	1/9/2012 10:38 AM	Rename  Unlink
Win 7	12/23/2011 9:32 AM	Rename  Unlink
Win 7	12/22/2011 10:47 AM	Rename  Unlink
Win 7	12/6/2011 10:07 AM	Rename  Unlink

Below the table, it states: 'You have 3 Windows computers, 2 Macs, and 0 Linux machines linked to Dropbox. [Add a computer.](#)'

Figure 15. Dropbox: List of registered computers

## Multiple Devices

It is possible to access a Dropbox account from different machines. After the installation of the Dropbox client, the user has to link the machine to the account by entering username and password, no additional activation is required. A list of all devices currently linked to the account is provided via the web interface (under Account → My Computers). This list shows the computer names, the time of last activity, and the IP address last used. Using the web interface, the user may rename and unlink computers (see Figure 15).

<sup>68</sup><https://github.com/driverdan/dropship>

<sup>69</sup><http://forums.dropbox.com/topic.php?page=2&id=37320#post-317090>

## Update Function

Dropbox has a high update frequency (sometimes as low as one week, see the release notes<sup>70</sup> for more details). Dropbox automatically updates the client software without any user interaction.

## Server Location

According to the Dropbox Help Center<sup>71</sup>, all files are stored on Amazon S3 servers in the United States. This has been confirmed by our analysis.








---

<sup>70</sup>[http://www.dropbox.com/release\\_notes](http://www.dropbox.com/release_notes)

<sup>71</sup><https://www.dropbox.com/help/7>

## 9. MOZY

### 9.1 Synopsis

Copy	Backup	Sync.	Sharing								
	✓			✓		✓	✓	✓	✓		
Reg.	Transport	Encryption	Sharing	Dedup.	Devices	Update	Location				
±	+	±	%	-	%	++	--				

### 9.2 Availability

Mozy<sup>72</sup> is a subsidiary of the EMC corporation<sup>73</sup> and is headquartered in Seattle, Washington<sup>74</sup>. Since April 2011, Mozy has been operated by VMWare<sup>75</sup> on behalf of EMC, however EMC retains the Mozy business.

**Operating Systems** The client application runs on Windows and Mac OS X. Furthermore, a web interface is provided which can be used to manage account information, download the Mozy software and request file restorations. Android and iPhone applications are also available<sup>76</sup>.

**Client Software Version** The client software used in our tests was version 2.4.3.0. The latest version is 2.10.3.0.

**Pricing** Mozy offers a free service providing 2 GB storage space (MozyHome Free). Additionally, a paid service for non-commercial users is offered (MozyHome). Customers can choose between 50 GB storage space accessible from one computer (\$ 5.99 per month) and 125 GB storage space accessible from up to three computers (\$ 9.99 per month). Additional storage can be purchased (20 GB of space for \$ 2.00 per month). Users can also add additional computers for \$ 2.00 per month per computer. Furthermore, Mozy offers a service for business users (MozyPro). Customers have to buy a license for each server (\$ 6.95 per month) and desktop (\$ 3.95 per month). There is no limitation of storage space but a minimum of 1 GB has to be purchased (\$ 0.50/GB per month).

<sup>72</sup><http://www.mozy.com>

<sup>73</sup><http://www.emc.com>

<sup>74</sup>All requests made to mozy.com will be redirected to mozy.de if the request is made from Germany.

<sup>75</sup>[http://www.theregister.co.uk/2011/04/05/vmware\\_gets\\_mozy/](http://www.theregister.co.uk/2011/04/05/vmware_gets_mozy/)

<sup>76</sup><http://mozy.com/mobile/>

**Account Termination** Mozy reserves<sup>77</sup> the right to terminate an account immediately and without notice if the user fails to renew subscription, fails to pay any fees or invoices when due or otherwise fails to comply with the services terms of usage.

**Certifications** Mozy is SOC 1 SSAE 16 Type 2 audited and ISO 27001 certified<sup>78</sup>.

## 9.3 Features

**Copy** Mozy does not provide a copy feature as defined in Section 2.1.1.

**Backup** Data can only be backed up using the client application. The client application provides additional functionality to back up locally. There is no specific drive or folder where data has to be placed for backup. By default, Mozy creates a list of different file categories such as music, photos, emails, documents. It then searches files on the hard disk and associates them with the relevant category automatically. These files are then backed up according to the schedule. Furthermore, users have the option to manually start the backup at any time (see Figure 16). Users can also create custom backup sets and select individual files, folders, or local drives to backup.

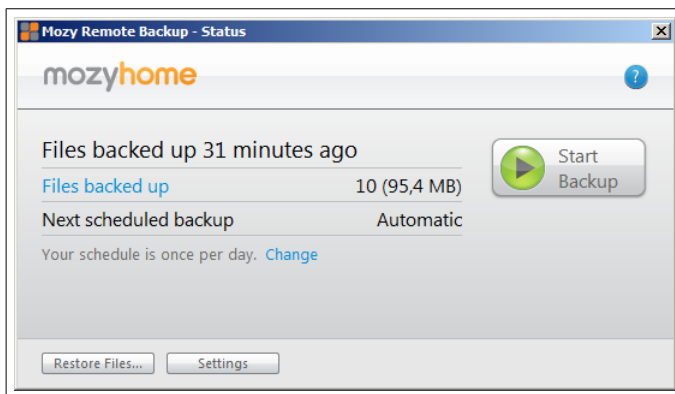


Figure 16. Mozy client application

Mozy keeps records of previous versions of a file and can restore to any version. The records are kept for 30 days, a very short time for a backup application.

<sup>77</sup><http://mozy.com/terms/>

<sup>78</sup><http://mozy.com/ssae16-iso27001>



Mozy offers to send a DVD containing the backup if transmission of data would take too much time.

**Synchronization** In the current client version, Mozy does not support synchronization of files between two or more computers attached to a single account. A synchronization feature (called Mozy Stash<sup>79</sup>) is currently in development<sup>80</sup>

**Sharing** Mozy does not support file sharing.

## 9.4 Security

### Registration and Login

Both the registration and the login process use secure communication channels (TLS). During the registration the user has to enter a name, an email address and a password (the user is prompted to repeat the password). If the email address is already in use by another Mozy account, the message “An account with that email already exists” is shown to the user. When registering for the free version of MozyHome, a CAPTCHA has to be solved before any feedback is given to the user (see Figure 17). The password has to be at least eight characters long, there are no other restrictions. Mozy gives no hint about the quality of the chosen password, weak passwords are accepted. In the MozyHome free registration process the user also has to provide additional personal data<sup>81</sup> to Mozy in order to complete the registration.

After registration, Mozy sends an email with a link which must be visited to activate the account. Then the user may use the web interface, or may download and install the client application. During the installation of the client application, the user is prompted to enter his email address and password. This information is stored locally, so it has to be entered only once.

To reset a password, the email address associated with the account has to be entered on the Mozy website and a CAPTCHA has to be solved. Then, an email is sent to the user which contains a secure link to a website where the user can set a new password. The new password has to be only six characters long. The old password is not changed unless the entire process has been completed by the user.

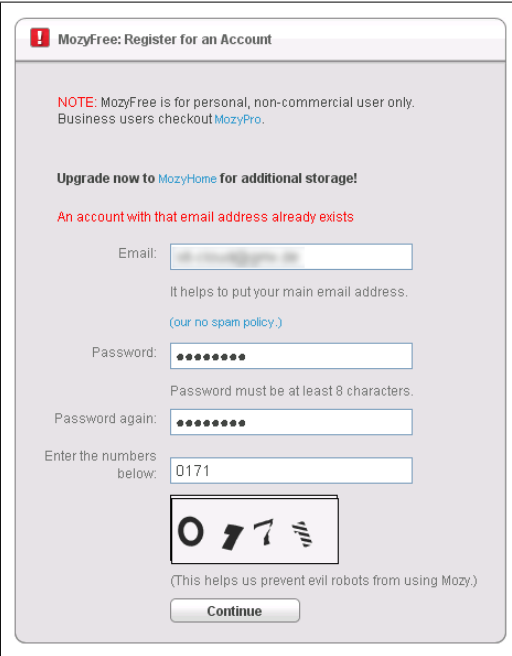
Mozy’s login process is not hardened against brute-force password attacks. In case of an incorrect login attempt, Mozy informs the user that one of the two is incorrect but not which one.

---

<sup>79</sup><http://mozy.com/backup/stash-beta/>

<sup>80</sup>Users can sign up for the beta version by logging into their Mozy account and opening [https://secure.mozy.de/account/stash\\_beta](https://secure.mozy.de/account/stash_beta)

<sup>81</sup>job category, name, primary role, zip code, gender, year of birth



**MozyFree: Register for an Account**

**NOTE:** MozyFree is for personal, non-commercial user only. Business users checkout [MozyPro](#).

**Upgrade now to [MozyHome](#) for additional storage!**

**An account with that email address already exists**

Email:


It helps to put your main email address.  
(our no spam policy.)

Password:

Password must be at least 8 characters.

Password again:

Enter the numbers below:



(This helps us prevent evil robots from using Mozy.)

Figure 17. Mozy: Registration process

## Transport Security

Mozy uses TLS to encrypt the communication between the client application and the server. The communication between the browser and the web interface is encrypted by using HTTPS.

## Encryption

All data is generally encrypted at the client, before being transferred to the Mozy server. The user can select between two encryption methods. The default is to use an 448-bit Blowfish encryption key which is provided by and therefore known to Mozy. As an alternative, the user can use a personal 256-bit AES encryption key. Unfortunately, Mozy points out only the drawbacks in choosing a personal key, that is the user has to be careful not to lose the key (see Figure 18). Mozy neither mentions the benefits of a personal encryption key nor the drawbacks of using Mozy's company key.

Users can restore their data using Mozy's web interface. When using this web restore function, the files that are to be restored are packed into a self extracting zip archive which may be downloaded by the user. When using Mozy's company key for encryption the archive contains



Figure 18. Mozy: Selection of the encryption method

the unencrypted files selected for backup. When using a private key, the zip archive provided by Mozy has to be decrypted on the client. This can be done by using the `MozyCryptoUtil`, which is provided by Mozy and will decrypt the restored files after the correct personal password has been entered.

Although Mozy encrypts the content of files uploaded to their servers, the file- and pathnames are stored unencrypted. This can be verified by setting up a Mozy account using a private key. When the web restore functionality described above is used, the downloaded zip archive contains the encrypted files with their original file- and pathnames. This does not meet our mandatory security requirements and therefore results in a downgrade.

## Sharing

Mozy does not offer to share files with other people.

## Deduplication

When using Mozy's company key for encryption, Mozy uses client-side cross-user deduplication. Mozy does not take any measures against the privacy issues concerning this deduplication method, i.e. Mozy does not use a threshold as described in Section 4.5. When a file is already on the Mozy servers and does not have to be uploaded by the user, a corresponding message is shown in the log of the Mozy client applicant, see Figure 19. The user does not have the option to

forgo deduplication. However, if the user decides to use a personal encryption key, cross-user deduplication does not work. In this case, Mozy uses client-side single-user deduplication.

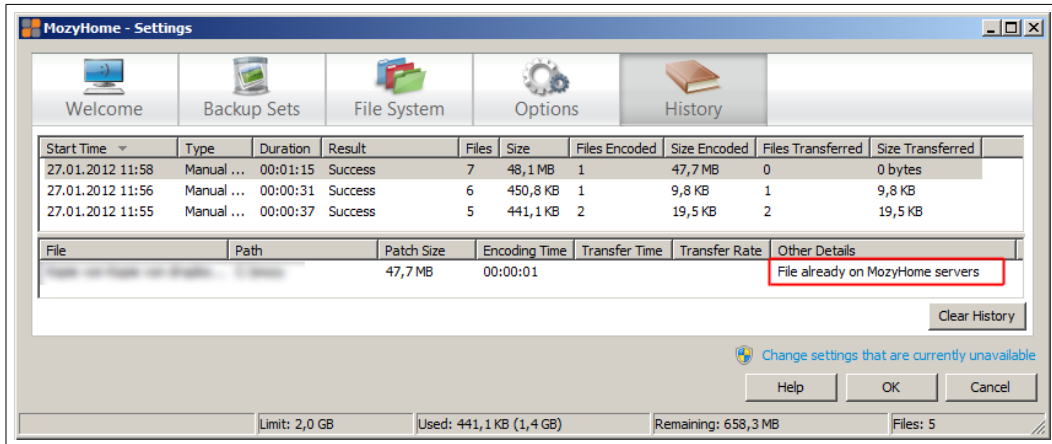


Figure 19. Deduplication in Mozy's client application

## Multiple Devices

A single Mozy account can be used to backup data from multiple machines. In the web interface a list of these computers can be displayed. For each computer, file restoration or deletion of the archive can be selected from this list. This feature is different from a synchronization scenario, because with Mozy the devices do not interact with each other.

## Update Function

In the client application, the user has the option to enable automatic updates without prompting the user to allow the update. Apart from that, the user can manually download and install updates from Mozy's website.

## Server Location

In its German data privacy statement<sup>82</sup>, Mozy states that its data centers are located in Europe. However, our analysis has shown that the Mozy client application uploads all files to servers in the USA. The zip archives used during the web restore function are also hosted on a server in

<sup>82</sup><http://mozy.de/datenschutz/verpflichtung/>

the USA<sup>83</sup>. According to information on the Mozy website, Mozy uses EMC infrastructure for storage.

The discrepancy between the advertised server location and the reality is odd. There are different sources available, in the German version of the security overview<sup>84</sup> no specific storage location is mentioned. However, in the same document it is claimed that all the Mozy data centers are located worldwide and that all data centers are Safe Harbor compliant, where applicable.

The apparent incorrectness of the server location stated by Mozy in its data privacy statement on the german website results in a downgrade.








---

<sup>83</sup>Our restoration archive was hosted at <https://dubl.mozy.com>

<sup>84</sup>[http://mozy.de/assets/631/A4\\_Mozy\\_Security\\_Overview\\_DE.pdf](http://mozy.de/assets/631/A4_Mozy_Security_Overview_DE.pdf)

## 10. TEAMDRIVE

### 10.1 Synopsis

Copy	Backup	Sync.	Sharing							
✓	✓	✓	✓	✓	✓	✓			✓	
Reg.	Transport	Encryption	Sharing	Dedup.	Devices	Update	Location			
±	±	+	±	%	±	++	++			

### 10.2 Availability

TeamDrive<sup>85</sup> is operated by TeamDrive Systems GmbH<sup>86</sup>, which is located in Hamburg, Germany.

**Operating Systems** Both the client application and the server application are available for Windows, Mac OS X and Linux. A web interface is also available, and can be used to upgrade licenses, to buy additional space and to manage the information associated with the account. The files stored on the servers can not be accessed through the web interface. Versions for Android and iPhone/iPad are currently being developed<sup>87</sup>.

**Client Software Version** The TeamDrive client software used in our tests was version 2.4.060. At the time of writing, the latest version available is version 2.4.161<sup>88</sup>.

**Pricing** TeamDrive follows the *freemium* business model also used by many other cloud storage providers. The basic service (TeamDrive Free) is free and provides 2 GB online storage space. Additionally, two premium services both providing 2 GB online storage space are available (TeamDrive Personal, 29.99 € per year and TeamDrive Professional, 5.99 € per month). The free and the personal versions have the same set of features, however the free version display a banner in the client and user of the free version only have access to limited support.

Some notable additional features in the professional version are LDAP synchronization, API support, publication of individual files through URL, email notification of comments to other team members, and free choice of version control. For the personal and premium versions, additional storage space can be purchased in increments of 10, 25, or 50 GB (starting from 5.99 € per month

<sup>85</sup><http://www.teamdrive.com>

<sup>86</sup><http://www.teamdrive.com/imprint.html>

<sup>87</sup><http://forum.teamdrive.net/viewtopic.php?f=8&t=235>

<sup>88</sup>Change log available at <http://mac.softpedia.com/progChangeLog/TeamDrive-ChangeLog-68387.html>

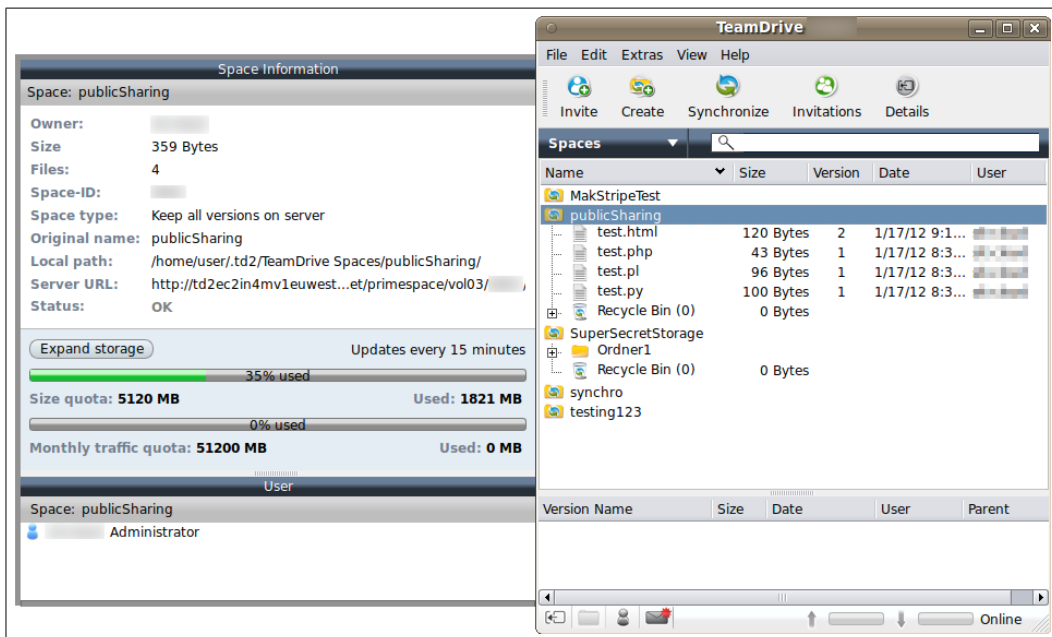


Figure 20. TeamDrive Gui – Space Overview

for 10 GB, with discounts for the larger amounts). The storage space for the free version can only be extended by using the referral system for a bonus of up to 8 GB.

TeamDrive also offers two server applications that enable users to host their own TeamDrive storage server. The TeamDrive Personal Server is aimed at small to medium-sized companies. There is a free version available, which is limited to 10 GB storage space on the server. Unlimited storage is available for 99.99 € per year. The TeamDrive enterprise server<sup>89</sup> is a scalable hosting server solution and is aimed at larger companies and Internet Service Providers. There is no pricing information available.

TeamDrive Bundles, a combination of server and client licenses are also available and offer a discount in comparison to individual purchase.

**Account Termination** TeamDrive Systems reserves<sup>90</sup> the right to terminate the contract with a period of notice of four weeks to the end of the month. Given good cause, e.g. when the customers' payments are more than 20 days late, TeamDrive (or the user) can terminate the account without

<sup>89</sup>[http://www.teamdrive.com/teamdrive\\_enterprise\\_server.html](http://www.teamdrive.com/teamdrive_enterprise_server.html)

<sup>90</sup><http://www.teamdrive.com/general-terms-and-conditions.html>

notice. Free users will be given four weeks to download all their files after they have been notified. For paid accounts, access to the data will be denied after the third request for payment has been sent to the user<sup>91</sup>.

**Certifications** TeamDrive uses Amazon Web Services for data processing and storage which is SAS 70 Type II certified. TeamDrive also offers data storage in a data center located in Hamburg, Germany, which is ISO 27001 certified. The TeamDrive software has been awarded the *Privacy Seal of Quality* (Datenschutzzügiesiegel) by the Independent Centre for Privacy Protection in Schleswig-Holstein, Germany<sup>92</sup>.

## 10.3 Features

**Copy** TeamDrive does not provide a copy feature as defined in Section 2.1.1, but the borderline to backup is blurred.

**Backup** TeamDrive introduces the concept of spaces. These spaces are similar to folders and can be created empty or based on an already existing folder. All files inside the spaces are kept on the server, new versions are transmitted automatically. For every space, an individual AES-256 key is used to encrypt the files.

Users can manually add files that should be backed up by creating a new space, by adding files to folders in the file system already watched by TeamDrive or by adding files directly inside the TeamDrive software. Deleted files can be recovered through the TeamDrive software.

To be able to restore files after a system failure, e.g. a hard disk crash, the user needs the backup copies of the key files. These key files (\* .pss) are created for every space and are stored on the users computer.

These files should also be stored in an alternative storage location like an USB stick. After reinstalling the TeamDrive software and logging in with the account credentials, the .pss files can be imported and the spaces will be restored from the server.

**Synchronization** TeamDrive supports synchronization between multiple computers. All files and folders that should be synchronized have to be added to a space. To sync this space to other computers, the corresponding TeamDrive installations have to be invited. This can be done by selecting the *Invite all my devices* option from the TeamDrive menu. Individual installations can not be invited, however by denying or accepting the invitations the user can configure to which devices he wants to synchronize his files.

---

<sup>91</sup> Additional information provided by TeamDrive upon request.

<sup>92</sup> <https://www.datenschutzzentrum.de/guetesiegel/kurzgutachten/g050302/>



If files are changed while the TeamDrive client is offline, the changes will be uploaded as soon as a connection to the server is established again. Conflicts are detected and a notification is displayed to the user. If the user selects “Resolve conflict”, the conflicting version will be shown (see Figure 21). From here it is possible to resolve the conflict by selecting one of the file as the current version. The user has to compare the files manually.

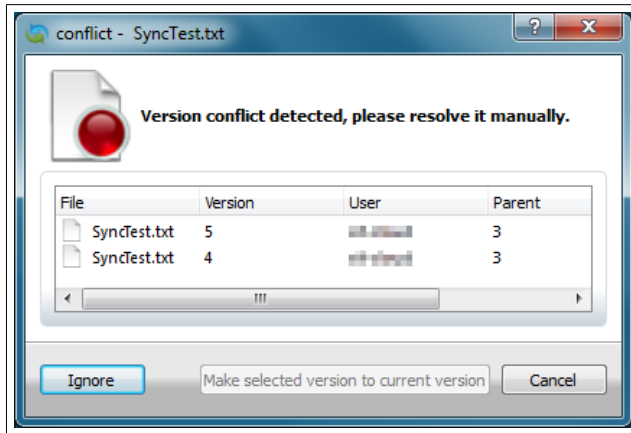


Figure 21. TeamDrive Synchronization Conflict

**Sharing** Files can be shared with subscribers of TeamDrive<sup>93</sup>. In addition, files can be shared with everybody. Sharing files with a closed group of non-subscribers is not supported, as it seems. TeamDrive should describe more clearly which flavor of sharing is meant.

- (1) *Sharing files with subscribers.* The inviting user downloads the public keys of the users he wants to invite from the TeamDrive server. Then the inviting user encrypts the invitation, which includes the AES key for the space, with the public key of the invited user and sends it via the TeamDrive server. The invited user receives the invitation, decrypts it with his private key and can access the space of the inviting user. It is possible to assign different permissions<sup>94</sup> when inviting other users: read, read/write, super user (can invite additional users), administrator (can remove users and delete files permanently from the space).
- (2) *Sharing files with everybody.* The Professional version of TeamDrive allows publication of files. These files are stored unencrypted on the TeamDrive server<sup>95</sup>.

<sup>93</sup><http://www.teamdrive.com/security.html>

<sup>94</sup><http://www.teamdrive.com/collaboration.html>

<sup>95</sup><http://www.teamdrive.com/collaboration.html>

Multiple TeamDrive users might be working on one file at the same time, which can lead to conflicting versions. In case of a conflict TeamDrive prompts an error message to the user that generated the conflict. The conflicting files are stored with the same version number but it is shown that they have been updated by different users. The conflict has to be resolved manually by selecting one of the two copies of the file as the current version (see Figure 22).

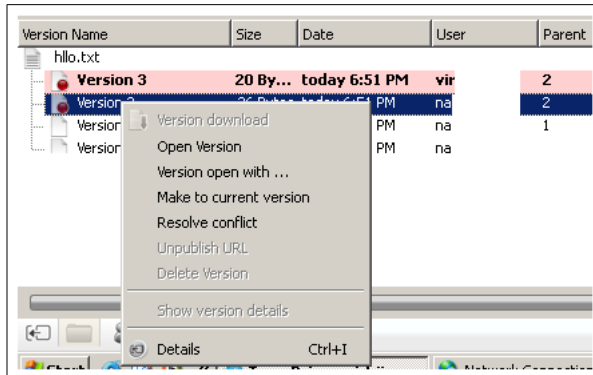


Figure 22. TeamDrive Conflict Resolution

## 10.4 Security

### Registration and Login

Both the registration and the login process on the website use secure communication channels (SSL/TLS).

For the registration, the user has to provide a user name, a password and a valid email address. The user name and email address have to be unique. The password has to be at least eight characters long, there do not seem to be any restrictions on what kind of characters can be used inside the password. Except for the mandatory length of eight characters, TeamDrive does not enforce any kind of password policy. After registration, an email containing a secure activation link (HTTPS) is sent to the email address associated with the account.

Every installation of the TeamDrive software has to be registered with an already activated user account. During the installation process, the user will be required to enter his user name and password. An activation email will be sent to the email address associated with the account. The installation procedure is paused until the user has successfully activated the new installation by clicking on the link in the email.

For every installation of the software, TeamDrive creates an individual public/private key pair. The public key is transmitted to and stored on the TeamDrive server. These keys are used to encrypt the messages exchanged between the server and the client and are also used to securely share files with other users.

If the user forgets his password a new password can be requested from the TeamDrive web page. After entering either user name or email address, an email is sent to the user which includes a link to a password reset form, where the new password can be entered. All websites accessed during the password reset process use secure communication (HTTPS)<sup>96</sup>.

The login process on the TeamDrive web page is currently not protected against brute force attacks. After notifying TeamDrive, they told us that this will be implemented in the near future. Both the login mechanism and the password reset function on the website can be used for information gathering. When trying to log in with a non-existing email address, a message “Username does not exist” is displayed. When using the password reset function with a non-existing email address or user name, a message “Username or email does not exist” is displayed.

TeamDrive enables information gathering regarding registered usernames and email addresses during the registration process and the password complexity required is not ideal. However, both the initial registration and every installation of the client have to be activated by the user which prevents the incrimination attack that has been mentioned before.

## Transport Security

The web interface uses HTTPS to secure the communication between browser and server. The communication between clients and the TeamDrive server uses HTTP, enhanced by a self-made, unpublished protocol. This is a violation of *Kerckhoff's principle*<sup>↑</sup>. As history shows, many protocols of this kind are weak. Therefore we downgrade TeamDrive in this category.

## Encryption

TeamDrive uses AES-256 for file encryption. The data is encrypted at the client before it is transmitted to the server. Every space uses an individual AES key for file encryption. These AES keys are not based on a password and are not known to TeamDrive, therefore TeamDrive is not able to access any data stored by its users on their servers.

---

<sup>96</sup>When we started our analysis, the password reset function was transmitting the new password in plain text over HTTP. After notifying TeamDrive of this issue it has been fixed immediately.

## Sharing

Sharing files is supported by cryptographic means. Disinviting of team members could be improved.

- (1) *Sharing files with subscribers.* When sharing files with another subscriber, the TeamDrive server sends the public key of the invitee. The inviting user encrypts the AES key of the space with this key. In doing so, he trusts that the received key is authentic (cf. Section 4.10.3, p. 49). An invitation including the encrypted space key is sent to the invitee. After decrypting the space key with his secret key the invitee can access and decrypt all files inside the space. However, there is problem. The invitee (project partner, colleague) receives the AES key of the space and in principal, probably by using some hacking tools, he is able to extract this key and to store it on his harddisk. Now, imagine the invitee is disinvited from the space. The AES key of the space is still the same, so excluding the former invitee from reading new files is implemented by traditional access control mechanisms, it is not longer based on the cryptographic strength of encryption. There are solutions for this problem, e.g. by introducing a new AES key every time a user has been disinvited from the space, but they are not implemented by TeamDrive. According to our security requirements (cf. p. 40) this problem yields in a downgrade.
- (2) *Sharing files with everybody.* After the user has selected the individual file for publication, the file is decrypted locally and transmitted to TeamDrive. The published file is hosted on the TeamDrive servers unencrypted and can be accessed via a URL. If the original file is changed afterwards the published copy is not affected. The URL contains the ID of the space and an obfuscated filename. There is no central overview of files that are currently being shared.

## Deduplication

TeamDrive does not use deduplication.

## Multiple Devices

It is possible to access a single TeamDrive account from different machines. As mentioned above, the installation on every device has to be activated by the user and is initially empty. Regarding the accessibility of the spaces, all additional installations are treated like a different user and therefore have to be either invited (`Invite all my installations` option inside the client) or the .pss files have to be copied to the second installation and imported through the client. There is no central overview of machines associated with a TeamDrive account and no way to remove other installations.

## Update Function






TeamDrive can be updated automatically. Updates can also be triggered manually from inside the TeamDrive software. If the automatic update is disabled, an email will be sent to the user notifying him when a new version is available. A detailed change log is available.

## Server Location

According to the information available on the website, TeamDrive stores all files in the EU. During our analysis, we could verify that TeamDrive uses the AWS eu-west-1 availability zone, the servers are located in Ireland. The TeamDrive data center is located in Hamburg, Germany.

## 11. UBUNTU ONE

### 11.1 Synopsis

Copy	Backup	Sync.	Sharing			X	iOS			
✓		✓	✓	✓	✓		✓	✓	✓	✓
Reg.	Transport	Encryption	Sharing	Dedup.	Devices	Update	Location			
++	+	--	++	+	±	++	-			

### 11.2 Availability

Ubuntu One<sup>97</sup> is operated by Canonical Ltd.<sup>98</sup>, which is located in London, UK.

**Operating Systems** A web interface is provided which may be used for account management as well as data access (see Figure 25). The client application is available for Windows and Linux. Android and iOS applications are available which support music streaming (using the Ubuntu One Mobile feature) and contact synchronization.

Since the service is being developed by the project lead of the Ubuntu project — Canonical Ltd. — and therefore first and foremost intended for Ubuntu users, additional integration directly into the operating system is provided. The menu bar’s indicator applet contains a shortcut to the Ubuntu One control center (see figure 26). Using the control center an account can be set up and data, Evolution<sup>99</sup> contacts, Tomboy<sup>100</sup> notes and Firefox bookmarks synchronization can be enabled (see Figure 23.) Additional integration into the system’s standard file browser Nautilus provides Ubuntu One configuration options from within the file explorer (see figure 24). The Ubuntu wiki has compiled a list of third party software<sup>101</sup> with support for Ubuntu One.

**Client Software Version** The client software is available for Ubuntu Linux 9.10 and higher, the tested version was *1.6.2-Oubuntu2*. For Windows XP or higher a client application is available with a reduced feature set. The tested client software version was *2.0.3*.

**Pricing** Ubuntu One uses the *freemium* business model which entails three different types of services. The Ubuntu One “Free” service can be used without cost and provides up to 5 GB storage space. To get more storage space, the service “Storage 20 GB” enables customers to

<sup>97</sup><https://one.ubuntu.com>

<sup>98</sup><http://www.canonical.com>

<sup>99</sup><http://projects.gnome.org/evolution/>

<sup>100</sup><http://projects.gnome.org/tomboy/>

<sup>101</sup><https://wiki.ubuntu.com/UbuntuOne/ThirdPartyProjects>

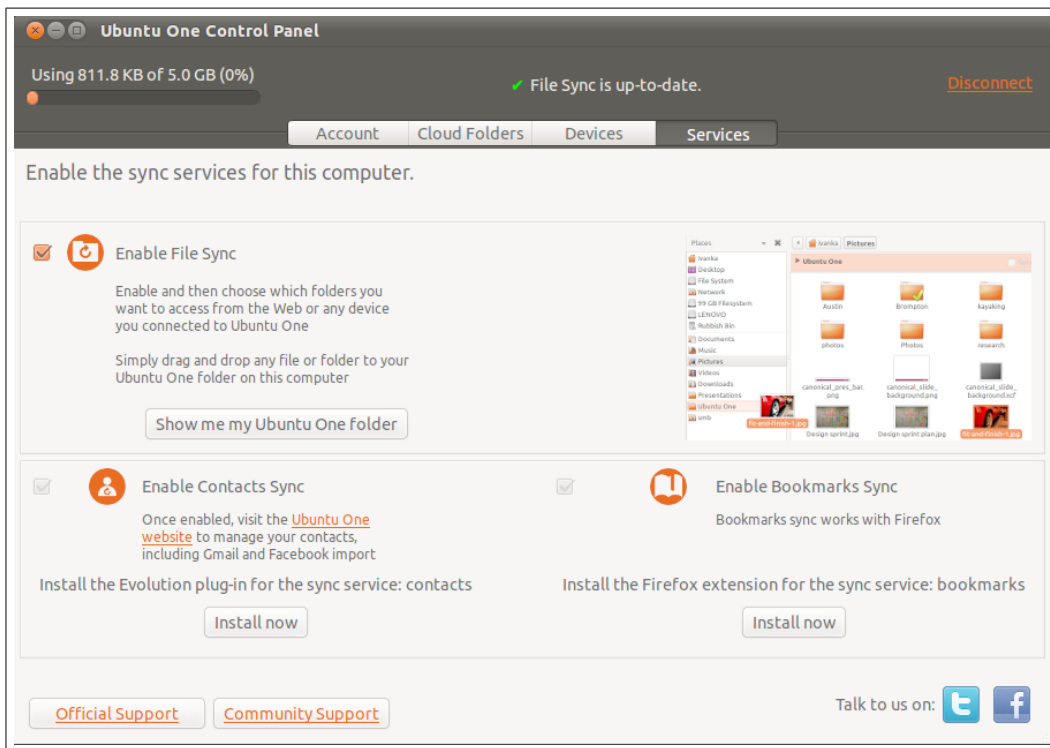


Figure 23. Ubuntu Linux 11.04: Ubuntu One Control Center

gradually add more space by means of 20 GB packs for a price of \$ 2.99 per month and pack or for an annual fee of \$ 29.99. No upper limit to the total number of packs is set. The additional feature “Music Streaming” adds 20 GB of storage, enables streaming music from the cloud storage directly to mobile devices and can be ordered at the price of \$ 3.99 per month or \$ 39.99 annually.

**Account Termination** Canonical’s Ubuntu One terms of service<sup>102</sup> state that the company may cease to offer the service if “commercially-practical” rates cannot be obtained. In this case, Canonical commits itself to inform all customers of impending service termination one month before the service ceases to exist. If any other reason leads to service termination, all customers are informed three months before the service is unavailable. If an account is inactive for a period of 90 days, Canonical reserves the right to delete any or all files after a notification email has been sent.

<sup>102</sup><https://one.ubuntu.com/terms/#your-account>

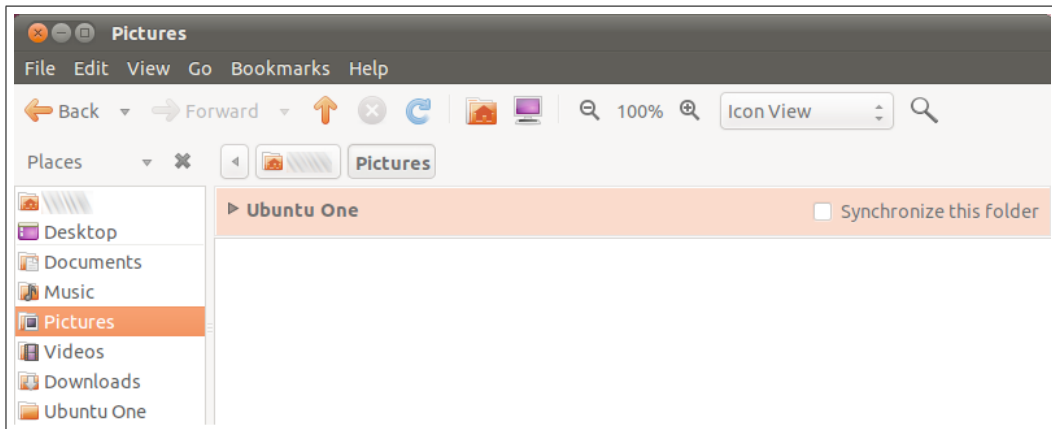


Figure 24. Ubuntu Linux 11.04: Ubuntu One Nautilus integration

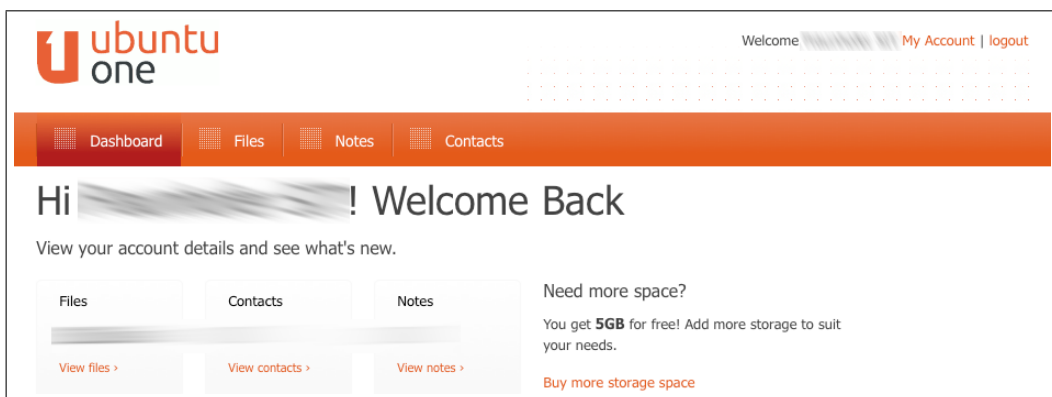


Figure 25. Ubuntu One: dashboard — account overview

On a side-note, the user experience with the software client is slightly different when comparing Ubuntu Linux and Windows implementations. While the Ubuntu linux client software comes with many direct integrations into the operating system and standard tools like the mail client Evolution, the Windows software — at the time of this writing — only supports file synchronization.



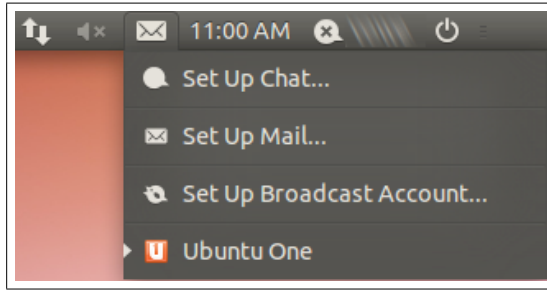


Figure 26. Ubuntu Linux 11.04: Ubuntu One tray icon

## 11.3 Features

**Copy** The standard folder used by the Ubuntu One service is hard coded to reside directly in the user's home folder and has the fixed name `Ubuntu One`. On Ubuntu Linux, integration with the file browser Nautilus allows choosing arbitrary folders to be synchronized in addition to the standard folder. The service starts after login to the system and automatically begins the uploading process.

A retrieval feature for previously deleted data is not available.

The Windows client does not yet offer the function to directly detect any changes made to files or directories. Thus, synchronization takes place on a schedule.

**Backup** Ubuntu One does not support a backup feature as defined in Section 2.1.2.

**Synchronization** Multiple machines can be connected to the service and all data is synchronized across all machines. Ubuntu One is able to recognize conflicts between two files on different machines, and resolves the conflict by deleting the file on the device where the conflict originated without notifying the user.

**Sharing** Files can be shared with subscribers or can be published. Sharing files with a closed group of non-subscribers is not supported<sup>103</sup>.

- (1) *Sharing files with subscribers.* The sharing of files with other Ubuntu One users is possible. Within the web interface, a click on the button `More` of any folder that the user wants to share brings up a context-menu where the option `Share on Ubuntu One` can be selected (see Figure 27). The service then displays all contacts which have been gathered in the internal address book. The user decides with whom the data should be shared and may grant read or read-write access. The internal address book can be synchronized with the Evolution mail

<sup>103</sup><https://one.ubuntu.com/help/faq/are-published-files-private/>

client in Ubuntu Linux or manually edited from within the web interface. Users can also decide to stop sharing any folder using the same context-menu.

All users who have been chosen to gain access to the shared directory receive an email with the information that the folder has been made available to them.



Figure 27. Ubuntu One: directory context menu

- (2) *Sharing files with everybody.* Beyond the sharing between registered users, directly publishing a file to the web is possible. In order to achieve this, clicking on the button `More` of any file that the user wants to publish and selecting the option `Publish file via Ubuntu One` will signal the system to publish the file (see Figure 28). Afterwards, the context-menu offers the option `Copy Web Link` — which will copy the file’s URL into the clipboard. The user is then responsible to make the URL known to all users with whom he wants to share the file. At any time, a user may choose to un-publish a file.

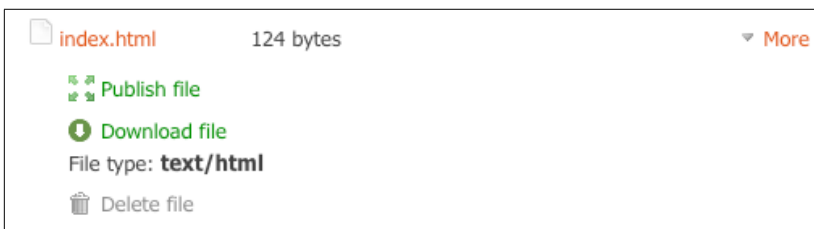


Figure 28. Ubuntu One: file context menu

**Technical Details** The Ubuntu One service operator Canonical Ltd. has compiled and released some technical details on the inner workings of their cloud storage service<sup>104</sup>. The Ubuntu One server application acts as a middleware between the client software and the Amazon S3 backend for actual storage. The custom protocol *u1storage* handles all communication between middleware

<sup>104</sup><https://wiki.ubuntu.com/UbuntuOne/TechnicalDetails>

and backend. It is based on Google's Protocol Buffers<sup>105</sup>. The reference implementation in Python has been released as open-source project under the AGPLv3 license<sup>106</sup>.

## 11.4 Security

### Registration and Login

Both the registration and the login process continuously use secure communication channels (SSL/TLS).

During registration, the user has to provide a full name (an arbitrary string), a valid email address, and a password which has to be repeated in order to recognize typos. Password rules demand a minimum of eight characters containing at least one number and one upper case letter. A password strength indicator helps with choosing a suitable password. Measure of choice to prevent automated account generation is the reCAPTCHA<sup>107</sup> system.

Ubuntu One does check whether the provided email address is already in use by another account but does not inform the user of this fact. Rather, an email is sent with information that a registration request has been undertaken using the email address. If the email is not already used by another Ubuntu One account, a six digit confirmation code is sent which has to be entered on the web site in order to activate the account. The system gives no information if the email address used during registration is already in use by another account, thus preventing information leakage such as username or email enumeration.

After registration, the user may use the web interface, or may download and install the client application. During installation of the client application, the user is prompted to enter the account credentials. On Ubuntu Linux, the password is stored in the operating system's keyring in encrypted form. If the keyring has the same password as the Ubuntu user account's password, the keyring is automatically unlocked when the user logs in. If not, the keyring password has to be entered when the Ubuntu One client attempts authentication to the online storage service.

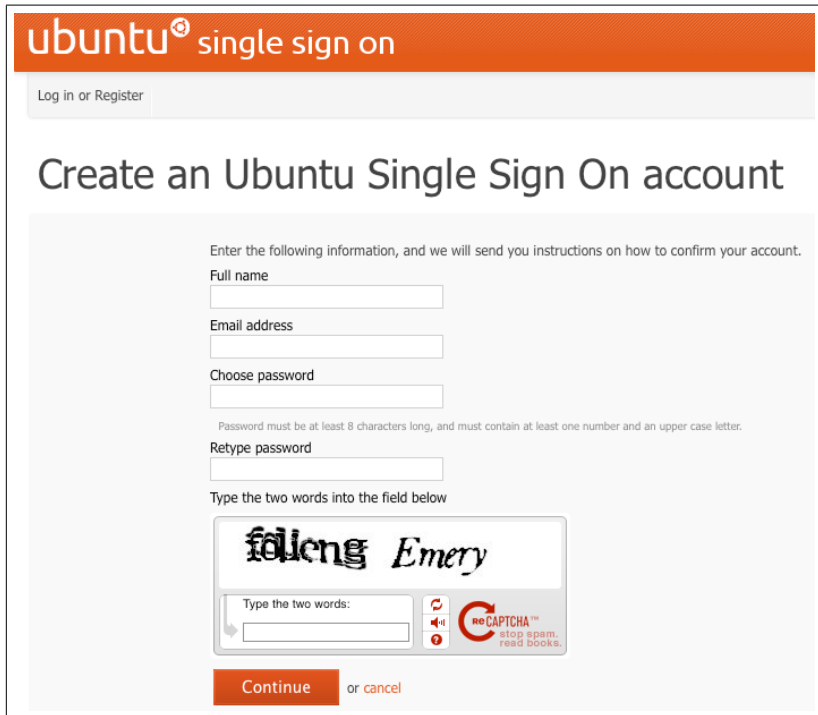
If the password is incorrectly entered multiple times during the login procedure, a temporary lock down of the account is enforced. This security measure aggravates brute-force attacks on the account credentials of the customers. If a user forgets his password, Ubuntu One sends an email to the address from which the user's account has been activated during registration. The email contains a six digit confirmation code to be entered on the website — the URL is also enclosed. When the confirmation code has been correctly entered, a new password may be chosen.

---

<sup>105</sup><https://code.google.com/apis/protocolbuffers/docs/overview.html>

<sup>106</sup><http://bazaar.launchpad.net/~ubuntuone-control-tower/ubuntuone-storage-protocol/trunk/files>

<sup>107</sup>reCAPTCHA™antibot system (<https://recaptcha.net>)



The screenshot shows the Ubuntu Single Sign On account creation interface. At the top, there is an orange header with the text "ubuntu single sign on". Below the header, there is a button labeled "Log in or Register". The main heading is "Create an Ubuntu Single Sign On account". Below this, there is a form with the following fields and instructions:

- Enter the following information, and we will send you instructions on how to confirm your account.
- Full name:
- Email address:
- Choose password:
- Retype password:
- Type the two words into the field below:

The CAPTCHA image shows the words "folong" and "Emery". Below the CAPTCHA, there is a text input field with the label "Type the two words:". To the right of the input field, there is a "no CAPTCHA" logo with the text "stop spam. read books." Below the form, there is a "Continue" button and a link "or cancel".

Figure 29. Ubuntu One: account creation

The registration process is cleverly designed, as no information gathering of email addresses is possible. The storage of the credentials to access the service — at least on Ubuntu Linux — into the internal keyring system can be regarded as exemplary.

Registration and login both meet our security requirements.

## Transport Security

Ubuntu One uses SSL/TLS to encrypt the communication between the clients and the server. The communication between the browser and the web interface is encrypted by using HTTPS. This meets our security requirements.

## Encryption

Ubuntu One does neither encrypt data using the client software nor on the server. Thus, the data itself is not protected against unauthorized access from attackers who successfully circumvent

authentication security of the service. Ubuntu makes the missing encryption very clear in their FAQ<sup>108</sup>. The missing encryption does not meet our security requirements.

## Sharing

- (1) *Sharing files with subscribers.* The sharing of files between registered Ubuntu One users meets our requirements.
- (2) *Sharing files with everybody.* The URL of a published file has the following format: `http://ubuntuone.com/h` — where *h* is a 22 character long random looking value. It consists of a mix of numbers and upper- and lower-case characters. The URL does not contain a username which impedes information gathering.

## Deduplication

Ubuntu One uses single-user deduplication which has no privacy issues. Apparently, deduplication is implemented on file-level instead of block-level. If small parts of a file are changed, the whole file is retransmitted to the server.

## Multiple Devices

A user may connect to the Ubuntu One service from different machines using the same account. The list of registered machines can be viewed using the web interface (see Figure 30). Apart from the host name of the connected device and the date when the machine has registered as Ubuntu One client, no further details are shown. The user has the opportunity to unregister any machine by removing it from the list.



Figure 30. Ubuntu One: connected devices listing

<sup>108</sup><https://wiki.ubuntu.com/UbuntuOne/FAQ/AreMyFilesStoredOnTheServerEncrypted>

## Update Function

The Ubuntu One client software is regularly updated for Ubuntu Linux systems<sup>109</sup>. Updates can be automatically installed via the integrated software repository. The client software for Windows operating systems has recently left beta status but offers a reduced feature set and has no built-in update function yet.

## Server Location

There is no information available on the Ubuntu One website regarding the location of the storage server. Ubuntu One uses US-based Amazon AWS to process and store all user data.








The Ubuntu One middleware handling the service's internal processes uses an EC2-instance in the US. It consists of a database where all file and directory metadata for every user are stored, including a pointer to the Amazon backend where the actual data storage occurs.

---

<sup>109</sup>Release Summary: <https://launchpad.net/ubuntuone-client/+download>

## 12. WUALA

### 12.1 Synopsis

Copy	Backup	Sync.	Sharing							
✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
Reg.	Transport	Encryption	Sharing	Dedup.	Devices	Update	Location			
-	±	±	±	-	-	++	+			

### 12.2 Availability

Wuala<sup>110</sup> exists since 2007 and is operated since March 2009 by LaCie AG<sup>111</sup>, which is located in Zurich.

**Operating Systems** Wuala is available for Windows (XP SP3, Vista, 7), Mac OS X (10.4+) and Linux (Java 1.5+). There is also a Java Application available<sup>112</sup> which can be used to access Wuala directly from the Web without installation. Mobile versions for iPhone, iPad and Android are also available.

**Client Software Version** Windows, Mac and Linux: We analyzed the *Witikon* version released on December 6<sup>th</sup> 2011. The new features introduced and changes to the previous versions are documented in a public changelog<sup>113</sup>.

**Pricing** Wuala uses the *freemium* business model which entails three different types of services. The *Free* service can be used without cost and provides up to 2 GB storage space. To get more storage space, the service offers three upgrade packages:

- 10 GB for annually 19 €
- 25 GB for annually 39 €
- 50 GB for annually 59 €

Additionally, Wuala offers a *Business* variant which encompasses 5 user accounts and 100 GB of storage space for an annual fee of 279 €.

<sup>110</sup><http://wuala.com/>

<sup>111</sup><http://www.lacie.com/>

<sup>112</sup><http://wuala.com/en/launch/>

<sup>113</sup><http://www.wuala.com/en/releasenotes>

**Account Termination** LaCie reserves the right to terminate the account for various reasons. In case of a violation of the Terms of Use<sup>114</sup>, the account can be deleted without notice. For other reasons like not accessing the account for 90 consecutive days or a discontinuation of the service, LaCie will provide at least 30 days prior notice of termination.

These rights apply regardless of whether LaCie is able to determine the content of the data.

## 12.3 Features

**Copy** Files can be added to Wuala only by using the client application. After login, files can be added by drag-and-drop or by selecting `Add files to Wuala` and will be uploaded immediately after addition. When files are deleted inside the Wuala application they are moved to the `Trash` folder. They can be restored either from there, or by using the “Time Travel” function, which allows the user to view (and restore) the contents of any folder for a specific time. The 10 most recent versions will be saved by Wuala. If a file or folder is deleted from `Trash`, there is no way to restore it again.

**Backup** Folders that should be automatically backed up to Wuala can be added via `Tools` → `Backup Overview`. After selecting a local folder that should be backed up, the user selects a folder in Wuala where the backup should be stored. By default a new folder with the same name as the local folder will be created. The backup interval can be configured (continuous, hourly, daily, weekly, monthly). Backups can also be started manually (when not using the continuous setting). Using a filter based on globbing patterns, the user can also exclude certain files (i.e. `*.jpg, img01*`) inside the folder from the backup. Log files for each backup folder are available.

**Synchronization** Multiple machines can be connected to the same account. Local folders that should be synchronized across multiple devices can be selected via `Tools` → `Sync Overview` and can be selected individually for each device. Log files for each Sync Folder are also available. Wuala detects conflicting versions of the same file on different local machines and tries to resolve the conflict. In case a file is simultaneously changed on multiple machines, Wuala creates a new file on each local machine containing the latest synchronized version from the other client. The file has the same name and a prefix informing about the conflicting state as shown in Figure 31. To successfully resolve the conflict, the versioning feature is helpful to track all changes leading up to the conflict.

**Sharing** There is no way to share individual files directly using Wuala. However, there are three different ways to share folders.

---

<sup>114</sup><http://www.wuala.com/en/about/terms>



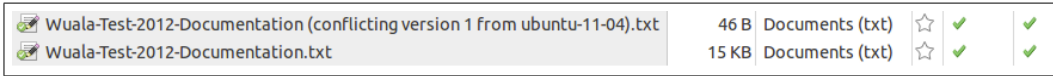


Figure 31. Wuala: file in conflicting state

- (1) *Sharing folders with subscribers.* Folders can be shared with contacts that are using Wuala. These can be individual contacts, or folders can be shared with All, including future contacts. There is no way to set individual permissions for the contacts, by default the invitees only have read access.
- (2) *Sharing folders with non-subscribers.* Individual folders can also be shared by using so called “Secret Weblinks”, which have the form `https://www.wuala.com/username/folder/?key=value`. The *value* can be set by the user (has to be at least 4 characters long), by default the *value* is random looking, 12 character long and consists of upper- and lowercase letters and numbers. To share the folder with other persons, the link has to be sent to them by the user using external communication channels (see Figure 32). There is no way to prevent the invitees from forwarding the link to additional persons, which is also mentioned in the FAQ<sup>115</sup>.
- (3) *Sharing folders with everybody.* When setting a folder’s visibility to `Public`, it will be visible to anyone from within Wuala and on the web and will be indexed by search engines (this is mentioned in the FAQ<sup>116</sup>). The public folders of users can be found at `http://wuala.com/username`.

The shared folders are highlighted in the Wuala Client using a combination of different colors (blue=public, red=shared) and icons, see Figure 33. The two options of shared are distinguished by different icons (contacts=head silhouette, Weblink = world map and lock).

Any sharing can be reversed by right clicking the folder and selecting `Make private` from the dropdown menu.

Additionally, Wuala allows the creation of groups which can be used to collaborate with other users. Similar to folder sharing, there are different kinds of groups, see Figure 34.

There are three different roles available: Member, Moderator and Administrator. For Member and Moderator, three different permissions can be configured: `Invite new members`, `Add and delete any item` and `Add items and delete them again`. The Administrator has full access, including the right to change the settings, change the permissions of the roles and members and remove members from the group.

<sup>115</sup><http://www.wuala.com/en/support/faq/c/20#id002017>

<sup>116</sup><https://www.wuala.com/en/support/faq/c/20#id002018>

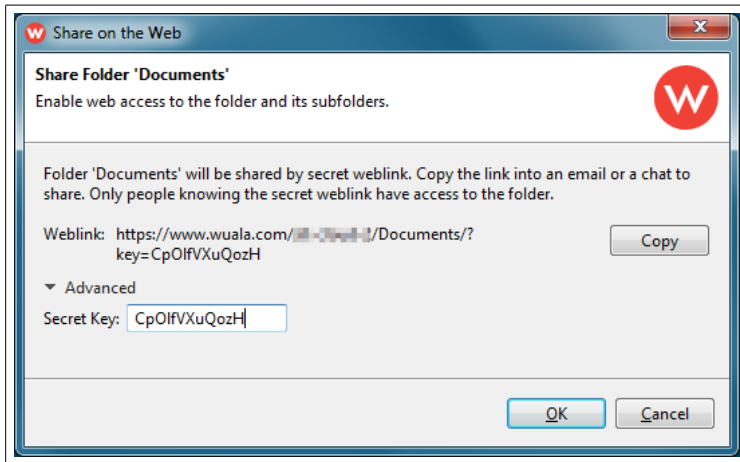


Figure 32. Wuala: Sharing with Secret Weblinks

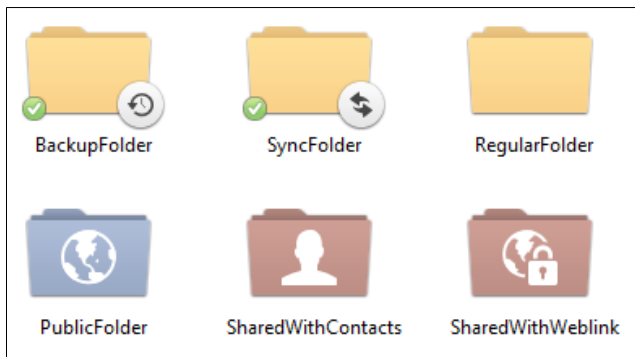


Figure 33. Wuala: Different folders

**Technical Details** Wuala has compiled an overview<sup>117</sup> of the technology employed by the service. While the propriety software parts are derivatives from implementations initially realized at ETH Zurich, Wuala has published parts of its core technology as Open Source projects:

- *Wuala Webstart*<sup>118</sup> is an application starter that directly loads required code and resources from a server.

<sup>117</sup><https://wuala.com/en/learn/technology>

<sup>118</sup><http://sourceforge.net/projects/wualawebstart/>

– *Wuala Persistent Map*<sup>119</sup> is a pure Java database optimized for use as a persistent hash map.

Additionally, Wuala has published<sup>120</sup> a comprehensive list of further Open Source projects and Third-Party code integrated into its client application.

---

<sup>119</sup><http://sourceforge.net/projects/quickbase/>

<sup>120</sup><https://wuala.com/en/about/thirdpartycode>

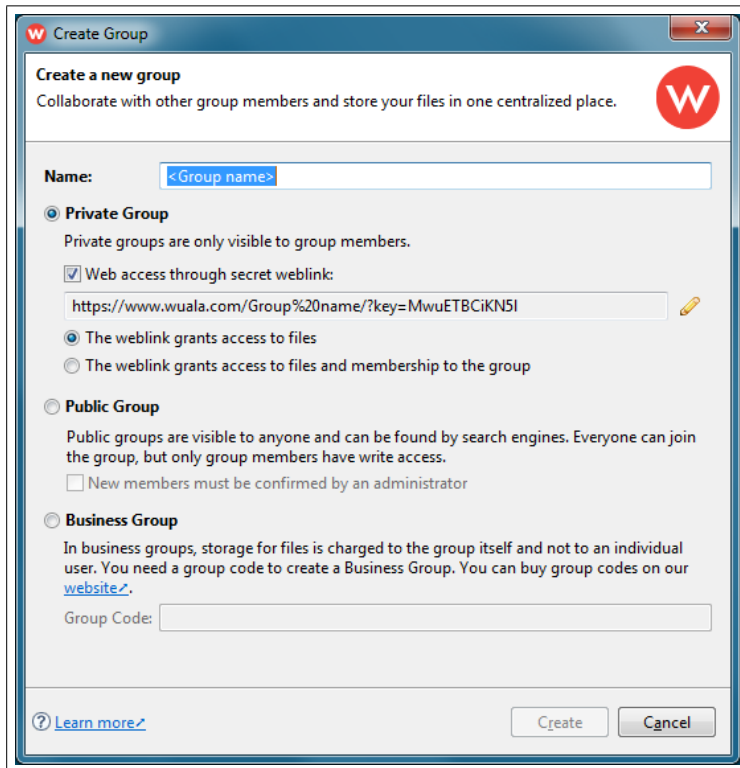


Figure 34. Wuala: Group creation options

## 12.4 Security

### Registration and Login

New accounts can only be created by using the Wuala application. During registration, the user has to provide a unique username, an email address and a password. The password has to consist of at least six characters, no other restrictions are enforced. Hence, Wuala allows weak passwords. Multiple usernames can be registered for the same email address. The registration does not have to be confirmed by the user, just a “Welcome to Wuala” message is sent to the email address used for the registration. In appendix B (p. 133) we have described an attack based on this weakness. Wuala has been informed.

Information gathering regarding already registered usernames is possible during the registration (This username is not available. Try a different one.). However, if the username does not already exist, a new account is created. There are other ways to find already registered usernames, see sharing.

Both the registration and the login process continuously use secure communication channels.

Since the passwords are not stored on the Wuala Servers, there is no way to recover a lost password. Wuala provides an optional “password hint” functionality, which can be defined either during the registration or any time later in the Preferences dialogue of the client. Password hints can be requested for a single username or for an email address and will be sent by email. If there are multiple accounts registered for one email address multiple emails will be sent, one for each account that has a password hint. The password hint function allows information gathering regarding already registered usernames and email addresses (see figure 35).

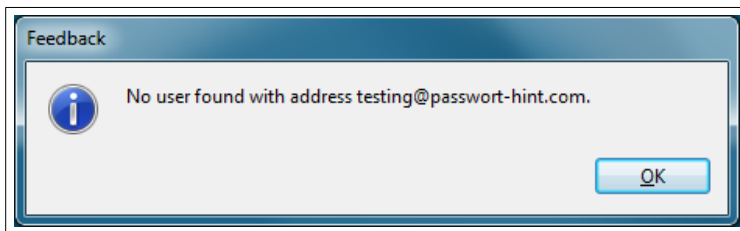


Figure 35. Wuala: Password hint feedback

There does not seem to be any restriction on failed logins.

## Transport Security

Wuala uses a proprietary client/server-communication protocol instead of the standardized and well-known SSL/TLS protocol to secure the communication between a client and the Wuala server. According to Wuala, integrity checks are used to protect transmitted data in transit but no detailed documentation about the mechanisms of its protocol have been published, a violation of *Kerckhoff's principle*<sup>↑</sup>.

In combination with the convergent encryption scheme (cf. next section) employed by Wuala, the absence of encryption during transmit allows attackers to sniff exchanged messages and attempt information gathering attacks.

## Encryption

The idea behind Wuala's encryption scheme is an untrusted file system that is secured by cryptographic methods. The employed system is an implementation of a folder tree structure for cryptographic file systems called *Cryptree* that has been published by Grolimund et. al. [GMSW06] from ETH Zurich. The trust anchor is a symmetric root key  $r$  which is derived from the user's password. Wuala calculates individual keys for every directory and individual keys for every file. All of them are accessible via  $r$ . They can be given to partners in order to share data.

Wuala uses *convergent encryption*<sup>121</sup> [DAB<sup>+</sup>02] to encrypt a file. That means, the key to encrypt a file is derived from its hashvalue. In the context of Wuala and slightly simplified this works in the following way (cf. Figure 36): Let *file* be the data to be encrypted, *fname* the name of the data on the user's disk and *s* a key generated at random. As mentioned, *s* is accessible via  $r$  by some cryptographic calculations. Let *enc* be a symmetric cipher and *hash* a cryptographic hash function. The client hashes *file* to derive a key  $k$  which is used to encrypt the contents of *file*. The

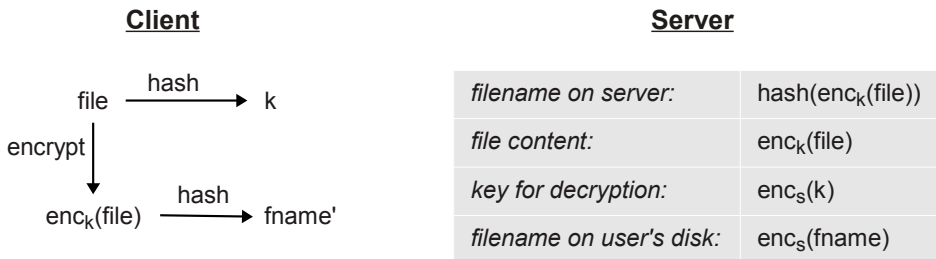


Figure 36. Encryption scheme of Wuala (simplified).

<sup>121</sup>Also known as *content hash keying*. Wuala uses the term *deterministic encryption*.

hashvalue  $fname'$  of this cryptogram, in some alphabetic representation, is later used as filename on the server's file system. After that, the following data is sent to the server<sup>122</sup>:

- (1)  $hash(enc_k(file)) = fname'$  the obfuscated filename
- (2)  $enc_k(file)$  the encrypted content
- (3)  $enc_s(k)$  the encrypted key that has been used for file encryption
- (4)  $enc_s(fname)$  the encrypted version of the real filename

The most important properties of convergent encryption are: (1) Identical clear texts are encrypted to identical crypto texts, independent of the user, (2) the server can not decrypt the crypto texts without having a copy of the clear text. The first property enables deduplication on encrypted data. The second property protects documents that are unique to a user, e.g. a self written, unpublished technical report. On the other side, convergent encryption has important drawbacks, in particular if an attacker has access to the server-side. The possible attacks include:

- (1) *Check for a file*<sup>123</sup>. Everybody who has the plain text can generate the cryptotext. Wuala knows this fact<sup>124</sup>. A server-side attacker can check whether a user has stored a certain file. To see the risk consider the following scenario: An attacker  $A$  having access to Wuala's filesystem wants to spy out the religious or political orientation of user  $B$ . For this purpose,  $A$  visits some public sites containing material (publications, pictures, manifestos) of the religious or political party he has in mind. By encrypting and hashing the material  $A$  can check whether  $B$  has stored this data on his computer. By repeating the attack with further material  $A$  can increase the soundness of his results.
- (2) *Disclosure of connections between users*. Consider  $A$  is a politician and  $B$  a journalist.  $A$  has given sensitive informations on a USB stick to  $B$ . Both of them are customers of Wuala, maybe without knowing this fact. After  $B$  has copied the file from the USB stick to his computer, an attacker on the server-side can detect that  $A$  and  $B$  share a file, hence the attacker presumes that they are working together in some sense. Please note, this attack works, even though  $A$  and  $B$  do not use the file sharing feature.

The service states that the password is never sent to its servers and thus no recovery is possible if the password is lost. However, a password hint can optionally be deposited as a last resort to recover a lost password. According to the release notes, Wuala uses AES-256 as of October 4<sup>th</sup> 2011 for metadata and storage encryption.

<sup>122</sup>Again, this is simplified. In the real implementation the meta data  $enc_s(fname)$  contain more than just the filename. Further, we ignore deduplication at the moment.

<sup>123</sup>We do not call this a *known-plaintext attack* because the objective of a known-plaintext attack is to reveal secret cryptographic parameters, e.g. the key used to encrypt the plain text. In our scenario there are no such parameters.

<sup>124</sup><http://bugs.wuala.com/view.php?id=3339>

The Wuala client signs every file using a key pair dedicated to the user in order to detect files that have been generated by an unauthorized party. Signatures are generated and validated using RSA-2048, while integrity checks use the SHA-256 hashing function.

Although the idea of a crypttree is sound, weakening the encryption by using deterministic keys in order to allow deduplication yields in a downgrade.

## Sharing

Security of shared files depends on the invitee (subscriber or non-subscriber).

- (1) *Sharing files with subscribers.* Sharing between registered users meets all mandatory requirements. The files are not readable by Wuala. When sharing files with another subscriber, the Wuala server sends the public key of the invitee to the inviting user. He encrypts a key for the invitee. The result is sent via Wuala to the invitee. In doing so, the inviting user trusts that the received keys are authentic (cf. Section 4.10.3, p. 49).

The sharing can be reversed for every shared item individually anytime. Wuala uses the concept of *lazy revocation*, introduced by Fu<sup>125</sup>. That means, if a user  $B$  is disinvited all existing files remain unchanged. This is based on the idea that  $B$  probably has seen these files. If a file is changed or a new file is added then these files are encrypted with new keys, not known to  $B$ . Hence, lazy revocation is a compromise between efficiency and security. From our point of view, lazy revocation is an appropriate concept.

Sharing in groups meets the additional requirement of configurable access rights.

- (2) *Sharing files with non-subscribers.* Sharing files with non sub-subscribers is based on secret web links, (e.g. <https://www.wuala.com/username/folder/?key=value>). Knowing the link is equal to having the right to access the file. The *value* included in the URL is sufficiently large, appears to be random and is only valid for this folder. The files shared with this method are not indexed by search engines, and sharing can be reversed anytime.

A secret web link contains both the username and the folder structure, which does not meet our requirements. All files shared with non-subscribers can be decrypted by the Wuala server<sup>126</sup>.

- (3) *Sharing files with everybody.* The link structure (<http://wuala.com/username>) used for the public sharing function allows enumeration of existing usernames even without signing up for Wuala, either by implementing a simple script or by using search engines (e.g. using Google: <http://www.google.de/search?q=username+site:www.wuala.com>).

---

<sup>125</sup><http://www.cs.umass.edu/~kevinfu/editorials/lazyrevocation.html>

<sup>126</sup><http://www.wuala.com/en/support/faq/c/20#id002017>



## Deduplication

Due to convergent encryption Wuala is able to perform client-side cross-user deduplication. Therefore, the client asks whether the obfuscated filename *fname'* (cf. Figure 36, p. 110) exists at Wuala. If this is the case, the transmission of the encrypted file content is omitted. Entire files are deduplicated, after changing the first byte of a file the entire file is uploaded again to Wuala. Wuala has the same privacy problems as services without encryption resp. services using a company key. Wuala does not use a threshold solution to avoid these problems.

## Multiple Devices

A user may connect to Wuala from different machines using the same account. There is no overview available regarding linked machines or devices and no way to remove other installations from accessing the Wuala account. For every synchronization folder, a list of participating devices can be displayed. However, only the currently used device can be withdrawn.

## Update Function

The Wuala client software is regularly updated<sup>127</sup>. Wuala checks for new updates on startup and multiple times per day<sup>128</sup>.

## Server Location

Wuala redundantly stores all files on servers in Switzerland, Germany and France<sup>129</sup>.

---

<sup>127</sup><http://www.wuala.com/en/releasenotes>

<sup>128</sup><https://forum.wuala.com/viewtopic.php?f=34&t=2493>

<sup>129</sup><https://www.wuala.com/en/support/faq/c/20#id002019>

## 13. SUMMARY OF FINDINGS

In the past sections, we found the following weaknesses:

- CloudMe** is open for a wide range of attacks, including username enumeration, sending unwanted emails, Cross-Side Request Forgery attacks, account hijacking and incrimination attacks.
- CrashPlan** uses a self-made, unpublished protocol for transport security, although SSL/TLS is an established alternative. It is not possible to remove individual installations.
- Dropbox** does not verify the email address at registration, hence it is open for incrimination attacks. Client-side encryption is not supported. It is unclear which flavor of sharing is used if non-subscribers are included (closed user group vs. publication).
- Mozy** encrypts files, but not filenames. The service does not manage cross-user deduplication in a secure way, thus enabling users to check if some file is already on Mozy's server. Weak passwords are accepted without notice.
- TeamDrive** uses a self-made, unpublished protocol for transport security, although SSL/TLS is an established alternative. It is not possible to remove devices again after they have been activated. When participants are removed from spaces, the cryptographic key used to encrypt the space is not changed. We also noticed that the password reset was using http transmission without any encryption, this has been fixed by TeamDrive after we notified them.
- Ubuntu One** does not provide any encryption.
- Wuala** does not verify the email address at registration, hence it is open for incrimination attacks. The service uses a self-made unpublished protocol, although SSL/TLS is an established alternative. The encryption scheme does not protect against attackers that have access to the unencrypted files. URLs shared with non-subscribers contain the user name.

# PART III: RECOMMENDATIONS AND CONCLUSION

## 14. LOCAL ENCRYPTION METHODS

The results of the cloud storage provider analysis made clear that most, but not all, cloud storage providers offer built-in methods to encrypt the data to be stored in the cloud. However, the encryption schemes are sometimes not sufficient, as some storage providers encrypt data by using an encryption key generated by and stored at the provider. This means that users cannot be sure whether the storage provider also uses the key to decrypt their data, access the contents and possibly pass it on to third-parties. Even worse, some cloud storage providers do not encrypt data at all.

While the absence of suitable client-side<sup>130</sup> encryption schemes may be criticized, a whole range of alternative methods are available which can be used to protect the data by locally encrypting them with keys only known to the user. This section will discuss some methods of encrypting data independent of the cloud storage provider, i.e. the user will have to use separate tools for encryption.

Figure 37 gives an overview of approaches. Each column presents one approach where the yellow box shows the encrypting instance. The names given in these boxes are just samples, of course there is always an alternative. The approaches can be divided in two groups: Approaches that hide all cryptographic data below the application level as given by the first to two columns and approaches that make cryptographic data visible on application level as shown shown by the boxes `enc/`, `enc-container` and `enc-doc` in the last three columns. Only the second group is helpful for users of cloud storage services. So we start with this group.

### 14.1 Approaches enhancing security

All of the next three approaches enhance the security of cloud storage users by encrypting data before transmission to the cloud. Although, the perfect solution is not recognized. This is due to insufficient integration into the cloud storage client, in addition some approaches have shortcomings wrt sharing encrypted data with other cloud storage users.

**Directory-based encryption** Directory-based encryption is a secure and simple solution to protect data in the cloud.

With directory-based encryption there are two directories, `enc/` and `clear/` (ref. Fig. 37). Data in `enc/` is encrypted but all of its files are mirrored in `clear/` as clear text. That means `clear/` is handled like a mounted version of `enc/`, and in fact often it is a mounted directory,

---

<sup>130</sup>Note on terminology: *client-side encryption* means a function of the software supplied by the cloud storage provider. *local encryption* means a method which is added by the user. Both of them are working on the user's side.

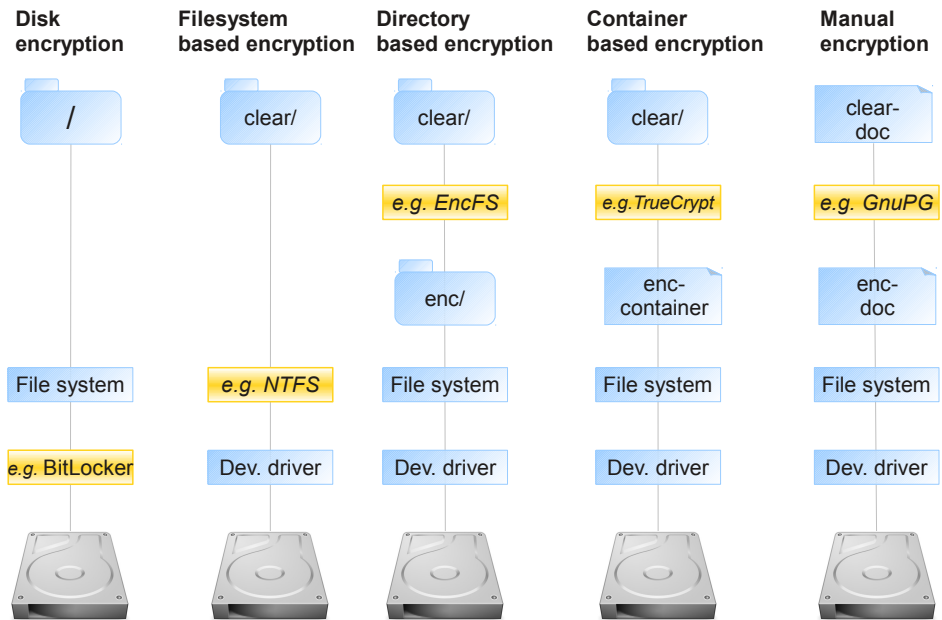


Figure 37. Local encryption methods.

with the additional feature of decryption which is activated by a password or some other credentials. The user can operate in a comfortable way on all files of `clear/`. If he modifies existing files or adds new files they are mirrored back to `enc/`.

Connecting `enc/` but not `clear/` to the cloud is a secure and simple solution to protect data. The drawback is that sharing files in `enc/` with other users is not possible because they don't have the key to decrypt the data. Another disadvantage occurs with storage providers that offer a transmission of file deltas. Imagine a big file made by a word processor. If one word is changed the difference of the old and the new version can be transmitted in a very efficient way by using delta encoding (cf. p. 23). However, after encryption the two versions are completely different so the whole file must be send to the storage provider.

Examples for tools working on mirrored directories are: EncFS<sup>131</sup> which is based on a file system in user space (FUSE) and works on top of any other filesystem. It is available for Linux, MacOS X and Windows. BoxCryptor<sup>132</sup> is a commercial software available for Linux, Windows,

<sup>131</sup><http://www.arg0.net/encfs>

<sup>132</sup><http://www.boxcryptor.com/>

MacOS X, iOS and Android that claims to be compatible with EncFS. eCryptfs<sup>133</sup> is an alternative to EncFS on Linux with enhanced cryptographic possibilities that allow sharing of data between hosts – in principle a way to share data between cloud storage users, but a very complicated one that is not really practical. SecretSync<sup>134</sup> is a free (for private users), closed source solution based on Java.

The lack of client-side encryption options of the popular cloud storage service Dropbox has spawned several helpful sources, including descriptions and setup of EncFS<sup>135</sup>, and TrueCrypt<sup>136</sup>. The public Ubuntu Linux forums provide a step-by-step explanation on how to encrypt all files prior to transmission to the Ubuntu One service using EncFS<sup>137</sup>.

**Container-based encryption** Container-based encryption is a secure solution to protect data in the cloud but it has some disadvantages compared to directory-based encryption.

The first step when using container-based encryption is to create the container, (`enc-container` in Figure 37) of a fixed size (e.g. 10 GB). After the container has been created, the software provides a way to mount the container into the underlying operating system, i.e. as a new drive (just like an external drive) by providing some credentials. Using this drive, mounted on `clear/` in our example, the user is able to store arbitrary files within the container and the software transparently handles decryption and encryption.

Connecting the container `enc-container` to the cloud, but not `clear/` is a secure solution to protect data.

Although, container-based encryption has some drawbacks. For example, assume the user has a 100 GB encrypted container containing many different files. Depending on the block-mode used for the encryption, changing only a few bytes of one of these files might change large portions of the container. This could result in the entire container being uploaded again. We did not analyze this in depth. In a short test we generated a container of 500 MB and added a file of 100 KB. Next time synchronization took place an amount of 140 KB was sent to the server. This is an overhead of 40% compared to sending the whole file.

Furthermore, synchronization may cause additional conflicts when using encrypted containers. If the container for instance contains two files, file *A* and file *B* and the user then changes file *A* with computer 1 and file *B* with computer 2, the encrypted container has been changed on both machines. The synchronization of the container will cause a conflict. If the two files had not been

---

<sup>133</sup><https://launchpad.net/ecryptfs>

<sup>134</sup><http://getsecretsync.com/ss>

<sup>135</sup><http://pragmaticca.wordpress.com/2009/05/10/encrypting-your-dropbox-seamlessly-and-automatically/>

<sup>136</sup><http://securosis.com/blog/how-to-encrypt-your-dropbox-files-until-dropbox-wakes-the-f-up>

<sup>137</sup><http://ubuntuforums.org/showpost.php?p=8512872&postcount=1>

stored in the same container and instead separately uploaded, no conflict would have occurred. In addition, when using encrypted containers it is not possible to share selected files within the container with other users.

An advantage of container-based solution is that complete containers can be shared with other users. If they can provide the credentials they can mount the container.

One of the most well-known tools for container based encryption is TrueCrypt<sup>138</sup>. TrueCrypt is a free open source tool available for Windows, Mac OS X, and Linux.

**Manual encryption** Individually encrypting every single file seems to be the most flexible way to locally encrypt data. Using this encryption method, storing files in the cloud will work as expected. Every single file will be separately uploaded to the cloud storage service. If the user changes one file, only this file has to be uploaded again. The drawback of this encryption method is that its handling may be quite cumbersome. The user has to actively encrypt and decrypt each file. After a file has been changed, it always has to be re-encrypted again. Furthermore, sharing files with other users has some implications: If the user wants to share such a file with other users, they must manage keys for encryption on their own, i.e. without support by the cloud storage client.

GnuPG<sup>139</sup> is the most prominent tool for manual encryption. It is a free open source implementation of the OpenPGP standard as defined in RFC 4880 [CDF<sup>+</sup>07]. It is available for Windows, Mac OS X, and Linux.

## 14.2 Approaches not enhancing security

We discuss two more approaches to local encryption: disk encryption and filesystem-level encryption. They are not adequate to enhance security of a cloud storage user because they encrypt data on a lower level of the operating system so that cryptographic data is never visible on application level. This implies that cloud storage clients always see the clear data, so this data is transmitted to the cloud. We describe these approaches for completeness and to prevent wrong understandings.

Please note, that “not enhancing security” is true from a perspective of cloud storage clients. In general it is recommended to use disk encryption or filesystem-level encryption in particular for every laptop because these solutions significantly enhance the security of the computer.

**Disk encryption** A new physical hard drive is normally unformatted and the available disk space needs to be split into one or several partitions. Disk encryption solutions provide encryption of all data on a whole partition. If a partition where an operating system resides is fully encrypted,

---

<sup>138</sup><http://www.truecrypt.org>

<sup>139</sup><http://www.gnupg.org>

the boot process fails unless the decryption key is available to decrypt all necessary data for the boot process. The decryption key is locked by a password chosen by the user and needs to be provided before access to the hard drive is possible. Thus, a so-called *pre-boot authentication* requires the password to unlock the key right after the computer's initialization.

Disk encryption is provided by professional software vendors and is also available on open source systems. In order to establish the pre-boot authentication process and also install hard drive driver software to transparently encrypt and decrypt all data, the installation of the disk encryption software deeply encroaches into the low level processes of the operating system and is mostly only available for a single type of operating system.

After the disk encryption has been successfully set up, the hard drive driver software automatically uses the key to encrypt and decrypt read and write operations on the disk transparently to the overlaying operating system. The data is thus only protected from unauthorized access if the computer or hard disk is stolen, since the password to unlock the key is not known. However, when working within the operating system, all data can be accessed by all applications, which includes the client applications of cloud storage providers. Disk encryption is therefore not suited to protect the user's data when using cloud storage.

Typical representatives of disk encryption are BitLocker for Windows, dm-crypt for Linux and FileVault for Mac OS X.

**Filesystem-level encryption** Each partition of a hard drive needs to be formatted before any data can be written onto it. The format process creates a file-system on the partition. A file system is an implementation of a set of rules that specify how the data is organized to enable efficient read and write operations. There are numerous different file systems in existence<sup>140</sup>, and they can be used by any operating system that either has built-in support, or otherwise a software driver is available. Naturally, all operating systems support at least one file system out of the box<sup>141</sup>. Some file systems have built-in support for encryption which can be used by all operating systems with a driver supporting it. Similar to disk-encryption schemes, the file system driver also enables transparent decryption and encryption to the overlaying operating system and is thus not suited to automatically transmit data in encrypted form to the cloud storage provider.

Filesystems that provide encryption include Microsoft's NTFS and reiserfs4 on Linux.

---

<sup>140</sup>For a general and technical overview over a range of file systems, see [https://secure.wikimedia.org/wikipedia/en/wiki/Comparison\\_of\\_file\\_systems](https://secure.wikimedia.org/wikipedia/en/wiki/Comparison_of_file_systems)

<sup>141</sup>among others Windows FAT and NTFS, Linux ext3 and ext4 and Mac OS X HFS



## 15. SELECTING A CLOUD STORAGE SERVICE

This study is not meant to nominate the best cloud storage service that fits all needs of any possible user. This is impossible. Instead, we want to give some advice, that may help selecting a service for a particular use case.

First of all, evaluate your use case, make clear, which problem you want to solve by using a storage service. Align your requirements to the features of the examined services, as given in table IV.

	Copy	Backup	Sync.	Sharing
CloudMe	✓			✓
CrashPlan		✓		
Dropbox	✓		✓	✓
Mozy		✓		
TeamDrive	✓	✓	✓	✓
Ubuntu One	✓		✓	✓
Wuala	✓	✓	✓	✓

Table IV. Features of Cloud Storage Services.

Second, consider:

- (1) How important is security for you?
- (2) How big is your amount of trust in other parties?

Having the answers, take a look at Figure 38.

Third, if you need more details than provided in the figure, then you will find in table V the summary of all grades of security related attributes that we assigned in this study:

The services can be sketched as follows:

- CloudMe provides a nice desktop, but from a security point of view it has many shortcomings.
- CrashPlan is focused on backup, provides client-side encryption, is unique in enabling the user to setup his own backup server. It has been downgraded wrt transport security because of using an unpublished, self-made protocol, although SSL/TLS is an established alternative that works fine.

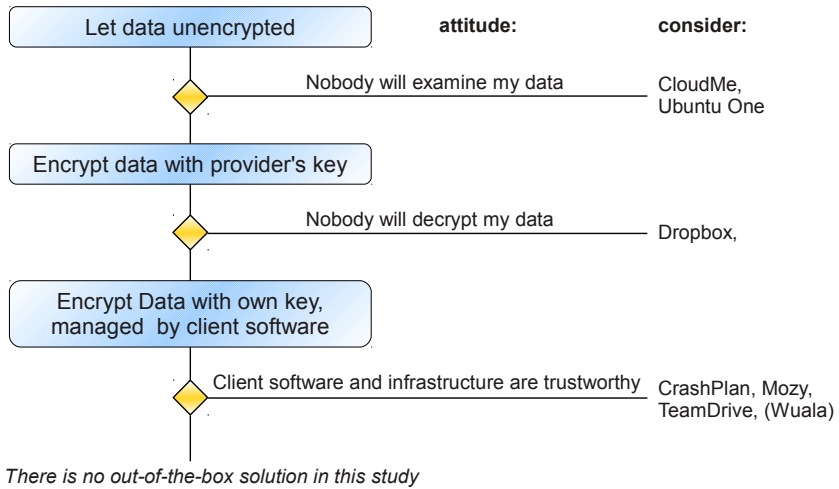


Figure 38. The right solution depends on the attitude of the user.

- Dropbox** has sophisticated synchronization features. It uses server-side encryption, and has therefore been downgraded wrt encryption.
- Mozy** is focused on backup, offers client-side encryption based on a managed key or based on a key that is known by the client only. The deduplication used by Mozy when using a managed key enables certain attacks.
- TeamDrive** is a full-featured (copy, backup, syncing, sharing), carries a data protection privacy seal, sharing data is supported by cryptographic means. It has been downgraded wrt transport security because of using an unpublished, self-made protocol, although SSL/TLS is an established alternative that works fine.

	Reg.	Transport	Encryption	Sharing	Dedup.	Devices	Update	Location
CloudMe	--	--	--	-	%	-	+	+
CrashPlan	+	±	+	%	+	±	++	++
Dropbox	-	+	-	±	+	±	++	+
Mozy	±	+	±	%	-	%	++	--
TeamDrive	±	±	+	±	%	±	++	++
Ubuntu One	++	+	--	++	+	±	++	-
Wuala	-	±	±	±	-	-	++	+

Table V. Grades. ++ very good, + good, ± some weaknesses, - bad, -- very bad, % not available

Ubuntu One is well integrated in Ubuntu Linux, but does not provide any kind of encryption.

Wuala is a full-featured (copy, backup, syncing, sharing) service. Due to convergent encryption it is open for attacks on encrypted data by a server-side attacker. Hence, Wuala is less secure than its competitors in the same category and written in parentheses in Figure 38. It has been downgraded wrt transport security because of using an unpublished, self-made protocol, although SSL/TLS is an established alternative that works fine.

If you have reached the bottom in Figure 38 (i.e. you did not agree with any service), then you may use one of the local encryption methods described in Section 14.1 (p. 116) to enhance a given service. Please note, some features may not work with these methods (as described there). It is easier to improve a simple service by local encryption methods than a complex one.

## 16. CONCLUSION

The study has defined mandatory security requirements for cloud storage providers that need to be fulfilled in order to be considered sufficiently secure, as well as some additional requirements that are not absolutely necessary. A few selected cloud storage providers have been analyzed and it was checked whether they meet these requirements.

As a major result, the study shows that most of the analyzed cloud storage providers are aware of the extreme importance of data security and privacy. Nevertheless, none of the examined cloud storage providers meets all mandatory security requirements. We discovered security threats regarding registration which should be easy to fix. Downgrading wrt transport security was caused by unpublished self-made protocols although SSL/TLS is an established alternative. Deduplication was a problem since two services did not use a well known threshold solution to prevent privacy attacks. Serious problems were caused by sharing files and by missing client-side encryption of data.

Regarding file sharing, there seems to be a discrepancy between the user's expectations (sharing files with a closed group) wrt privacy of information stored online and the implementation of the provider (publication of files for everybody). Providers should make very clear what is meant by *sharing files*. On multiple occasions, we were able to access information that was clearly supposed to be accessible only to a closed user group. In some cases, this information could be found via search engines. Sometimes, links for accessing shared data had not been obfuscated, or it was possible to iterate through URL combinations. An open problem occurs if subscribers and non-subscribers want to share encrypted data without revealing the contents to the provider.

Regarding client-side encryption of data the study shows very heterogeneous results. Some services do not use client-side encryption or use a company key, owned by the provider. Users do not hesitate to subscribe to such a service as the success of Dropbox shows. We suppose accepting the drawbacks of reduced security is acceptable in favor of simple usage as having access to data from anywhere from any device without key management. From a provider's point of view this an appreciated situation because there is a trade-off between security and costs: Usage of company keys for encryption or omitting encryption has the effect that data deduplication can be performed effectively which decreases the amount of storage capacity needed by the provider.

Convergent encryption as a means to enable deduplication on encrypted data is a compromise with handicaps, in particular it enables Check-for-a-file-attacks by a server-side attacker.

Using a personal key to encrypt data before transmission to cloud storage services is mandatory for business users. The confidentiality of business data has the highest priority as unauthorized disclosure may lead to economic ruin. The number of services providing a personal key in this study equals the number of services without this feature.

If a service does not support client-side encryption out of the box, additional tools like TrueCrypt or GnuPrivacyGuard can perform local encryption. However, not all available methods for local encryption are perfectly suited for a usage in connection with cloud storage. For instance, container-based encryption may cause higher network traffic, and yields additional conflicts when synchronizing data stored on different computers. Moreover, using tools made without cloud computing in mind will require an extra effort by the user as they have to be installed and the keys need to be managed by the user.

Even if using client-side encryption, the user should be aware that he trusts the provider by using-client-side software supplied by the provider. As cloud storage services collect a lot of data, they may be an interesting target for any kind of espionage.

In addition to concrete security requirements it is recommendable to observe some extra aspects. It is worthwhile to consider using more than one service to reduce the impacts of service downtime. Further, calculation of the time to recover all data from the cloud is recommended. Depending on the individual amount of data, this may take several days. Having a plan for a provider change in the future reduces the dependency on a particular provider (*provider lock-in*). This will be relevant, for example, if the chosen provider is getting to expensive or is not longer compliant with governmental rules.

The study also addresses the legal aspects of using cloud storage services. Both German and international legal regulations have been considered. To sum up, the legal implications and problems when processing (personal) data in the cloud are not yet sufficiently addressed and solved. As long as there are no consistent international regulations regarding cloud-based data processing, the data should remain within the EEA. Even here, it has to be considered that the cloud storage provider may be subject to different legal regulations than the user himself. Currently, using a purely Europe-based company to store data seems to be the only way to guarantee an adequate level of privacy protection.

## Glossary

*Application Programming Interface (API).* An API is a documented set of rules that can be encoded into applications in order to provide technical functionality which can be used for integration into external software. Often, access to APIs is used to offer digital services over public networks and standardized communication protocols are used. 15, 21

*Brute-force attack.* A brute-force attack is, as its name implies, no sophisticated attack. Rather, all possible combinations such as username/password pairs are tried out one after the other until a valid combination is found. Since authentication systems on public networks are directly accessible, these attacks possibly affect every service available to the general public. Brute-force attacks are often conducted in a massively-parallel manner — by trying out a huge number of possible combinations all at once on multiple machines. 37, 126

*Credentials.* In the scope of information technology, credentials are used to control access to information or services. The credentials exist in different forms on public networks such as username/password pairs or certificates. Several systems use biometric patterns (such as fingerprint or retinal patterns) as credentials. 36

*Dictionary attack.* A dictionary attack is a more sophisticated version of the *brute-force attack*<sup>↑</sup>. The attack tries out a list of the most popular passwords or uses all existing words from a dictionary in order to guess passwords. Since many individuals tend to choose easily memorizable passwords — which often represent already existing words, dictionary attacks can be very effective. 37

*Hash function.* A hash function is a mathematical function that maps data of arbitrary size onto values of (generally) fixed size. These functions should be easy to compute, but it should be computationally infeasible to construct a message that maps onto a given hash value or to find two different messages with the same hash value. Additionally, even small changes in the input should have a strong effect on the computed hash value. 126

*Hash value.* Result of a *hash function*<sup>↑</sup>. 22, 23

*Kerckhoff's principle.* A cryptographic principle, named by the cryptographer Auguste Kerckhoff (1835 – 1903), which says that a cryptosystem should be secure even if everything about the system, except the key, is public knowledge. The background is that every secret is a potential point of failure. In contrast, a public system can be discussed by a multitude of experts, which will find weaknesses, if there are any. There are some prominent examples of algorithms which violated Kerckhoff's principle, hence they have been broken. Included are the algorithms A5/I and

*A5/2*, used for GSM authentication, as well as *crypto-1*, used by the MIFARE chip. Kerckhoff's principle is broadly accepted by cryptographers around the world. 38, 68, 91, 110

*Social engineering*. Gaining access to systems or data by exploiting human psychology, rather than by breaking in or using hacking techniques . 38

*Software Development Kit*. A software development kit usually consists of basic implementations of the covered functionality in several popular programming languages. It may also include examples and relevant documentation. 22

## ACKNOWLEDGEMENTS

The authors thank Sven Bugiel, Matthias Enzmann, Markus Schneider and Annika Selzer for helpful comments.



## REFERENCES

- [AIC09] AICPA. Service Organizations, Applying SAS No. 70, as Amended, 2009.
- [AWV95] AWW. Arbeitsgemeinschaft für wirtschaftliche Verwaltung e.V., Grundsätze ordnungsmäßiger DV-gestützter Buchführungssysteme (GoBS), November 1995. [http://www.bundesfinanzministerium.de/nn\\_314/DE/BMF\\_\\_Startseite/Service/Downloads/Abt\\_\\_IV/BMF\\_\\_Schreiben/015,templateId=raw,property=publicationFile.pdf](http://www.bundesfinanzministerium.de/nn_314/DE/BMF__Startseite/Service/Downloads/Abt__IV/BMF__Schreiben/015,templateId=raw,property=publicationFile.pdf).
- [BfD10] BfDI. Der Bundesbeauftragte für den Datenschutz und die Informationsfreiheit, Federal Data Protection Act (BDSG), June 2010. <http://www.bfdi.bund.de>.
- [BMF01] BMF. Bundesministerium der Finanzen, Grundsätze zum Datenzugriff und zur Prüfbarkeit digitaler Unterlagen (GDPdU), July 2001. [http://www.bundesfinanzministerium.de/nn\\_95356/DE/Wirtschaft\\_\\_und\\_\\_Verwaltung/Steuern/Veroeffentlichungen\\_\\_zu\\_\\_Steuerarten/Abgabenordnung/Datenzugriff\\_\\_GDPdU/node.html?\\_\\_nnn=true](http://www.bundesfinanzministerium.de/nn_95356/DE/Wirtschaft__und__Verwaltung/Steuern/Veroeffentlichungen__zu__Steuerarten/Abgabenordnung/Datenzugriff__GDPdU/node.html?__nnn=true).
- [BMF11a] BMF. Bundesministerium der Finanzen, Abgabenordnung (AO), April 2011. [http://www.gesetze-im-internet.de/ao\\_1977/index.html](http://www.gesetze-im-internet.de/ao_1977/index.html).
- [BMF11b] BMF. Bundesministerium der Finanzen, Handelsgesetzbuch (HGB), March 2011. <http://www.gesetze-im-internet.de/hgb/index.html>.
- [BSI11] BSI. Bundesamt für die Sicherheit in der Informationstechnik, Security Recommendations for Cloud Computing Provider, 2011. <http://www.bsi.bund.de>.
- [CDF<sup>+</sup>07] J. Callas, L. Donnerhackle, H. Finney, D. Shaw, and F. Thayer. OpenPGP Message Format. RFC 4880, November 2007.
- [Con08] Chris Connolly. The US Safe Harbor - Fact or Fiction?, 2008.
- [Cou09] David A. Couillard. Defogging the Cloud - Applying Fourth Amendment Principles to Evolving Privacy Expectations in Cloud Computing, 2009. [http://www2.tech.purdue.edu/cit/Courses/cit556/readings/Couillard\\_MLR.pdf](http://www2.tech.purdue.edu/cit/Courses/cit556/readings/Couillard_MLR.pdf).

- [DAB<sup>+</sup>02] John R. Douceur, Atul Adya, William J. Bolosky, Dan Simon, and Marvin Theimer. Reclaiming Space from Duplicate Files in a Serverless Distributed File System. Technical report MSR-TR-2002-30, Microsoft Corporation, July 2002. <ftp://ftp.research.microsoft.com/pub/tr/tr-2002-30.pdf>.
- [DR08] T. Dierks and E. Rescorla. RFC 5246 - The Transport Layer Security (TLS) Protocol Version 1.2, August 2008.
- [EHG<sup>+</sup>11] Jens Eckhardt, Marc Hiber, Rüdiger Giebichenstein, Fabian Neumann, Thomas Helbing, and Andreas Weis. Guidelines Cloud Computing - German Law, Data Protection and Compliance, 2011. <http://en.eurocloud.de/2011/03/04/eurocloud-guidelines-cloud-computing-german-law-data-protection-and-compliance/>.
- [Fie00] Roy Thomas Fielding. *Architectural Styles and the Design of Network-based Software Architectures*. PhD thesis, University of California, Irvine, 2000.
- [GMSW06] D. Grolimund, L. Meisser, S. Schmid, and R. Wattenhofer. Cryptree: A folder tree structure for cryptographic file systems. In *Reliable Distributed Systems, 2006. SRDS '06. 25th IEEE Symposium on*, pages 189–198, oct. 2006.
- [HNM<sup>+</sup>03] Marc Hadley, Henrik Frystyk Nielsen, Noah Mendelsohn, Martin Gudgin, and Jean-Jacques Moreau. SOAP version 1.2 part 1: Messaging framework. first edition of a recommendation, W3C, June 2003. <http://www.w3.org/TR/2003/REC-soap12-part1-20030624/>.
- [HPSP10] D. Harnik, B. Pinkas, and A. Shulman-Peleg. Side Channels in Cloud Services: Deduplication in Cloud Storage. *Security Privacy, IEEE*, 8(6):40–47, nov.-dec. 2010.
- [IDW10] IDW. Institut der Wirtschaftsprüfer in Deutschland e.V., Prüfung des internen Kontrollsystems beim Dienstleistungsunternehmen für auf das Dienstleistungsunternehmen ausgelagerte Funktionen, November 2010.
- [ISO05] ISO/IEC. Information Technology - Security Techniques - Information Security Management Systems - Requirements, December 2005. <http://www.iso.org>.
- [MG11] P. Mell and T. Grance. The NIST Definition of Cloud Computing (Draft), January 2011. <http://csrc.nist.gov/publications/drafts/800->

[145/Draft-SP-800-145\\_cloud-definition.pdf](#).

- [MS11] Ninja Marnau and Eva Schlehahn. Cloud Computing und Safe Harbor. *DuD*, 5, 2011.
- [MSL<sup>+</sup>11] Martin Mulazzani, Sebastian Schrittwieser, Manuel Leithner, Markus Huber, and Edgar R. Weippl. Dark Clouds on the Horizon: Using Cloud Storage as Attack Vector and Online Slack Space. In *USENIX Security*, 8 2011.
- [New11] Derek Newton. Dropbox Authentication: Insecure by Design. <http://dereknewton.com/2011/04/dropbox-authentication-static-host-ids>, April 2011.
- [Sog09] Christopher Soghoian. Caught in the Cloud: Privacy, Encryption, and Government Back Doors in the Web 2.0 Era. *8 J. on Telecomm. and High Tech. L.* 359, 2009. <http://ssrn.com/abstract=1421553>.
- [Wei10] Thilo Weichert. Cloud Computing und Datenschutz. *DuD*, 10, 2010.

## APPENDIX A

### ATTACK ON CLOUDME DESKTOP

The analyzed web application of CloudMe exhibits multiple vulnerabilities:

- (1) The services provides a SOAP-based API enabling remote access to web application capabilities. The function `queryUserDB` features information leakage and thereby enables attackers username enumeration by supplying the parameter “\*” as search term.
- (2) The API functions `createDocument`, `copyDocument` and `deleteDocument` can be used to create email conform documents which can subsequently be sent through the internal messaging system of the web application. It enables attackers to embed HTML and JavaScript code into documents and send them to other users. The email’s subject and the sender’s email address may be arbitrarily chosen.
- (3) The internal message viewer of the web-based desktop allows HTML display and Javascript code execution of inbound emails, thereby facilitating Cross-Side Request Forgery (CSRF) attacks.
- (4) The web-based desktop features a flaw within the account authentication system. It provides a function to set a new password for the user’s account without requiring the currently used password. These vulnerabilities can be combined into an account hijacking attack:
  - The attacker first gathers the account usernames through the SOAP-based API
  - Using the gathered username information, a spear phishing attack can be launched: The attacker sends an unsolicited email containing JavaScript code which changes the user’s account password to each user. To ensure a high probability that the recipient opens the email in the internal viewer the message subject and the sender’s address can be set to appear as originating from the CloudMe administrative staff (i.e. `admin@cloudme.com`).
  - As soon as the victim views the email with the internal viewer, the JavaScript code is executed and conducts the CSRF-attack to directly change the account’s password.

An optimized attack could employ further JavaScript code in order to send a message back to the attacker informing him of the password change of a specific user account.

CloudMe has been informed about the problems.

## APPENDIX B

# INCRIMINATION ATTACK ON CLOUDME, DROPBOX AND WUALA

On registration a cloud storage service should verify that the email address of the user really belongs to that user. A simple notification email which does not require any further step is not suitable to provide sufficient security.

Below we describe an attack on CloudMe, Dropbox and Wuala that is based on missing email verification. In the scenario, an attacker is using a cloud storage service to store illegal or incriminating files in the name of his victim. It is necessary, that the victim is no registered customer of the corresponding service yet. We assume that the attacker knows a valid email address of his victim. The attack consists of the following steps:

- (1) The attacker registers an account using the name and email address of his victim.
- (2) The attacker uploads incriminating material using the account of his victim to the cloud storage service. This can be illegal content, e.g., pictures with child abuse.
- (3) The attacker notifies authorities, e.g., the police, about the illegal and incriminating material.

Instead of notifying authorities, it is also possible to incriminate the victim at friends and colleagues. This might have unpleasant consequences for the victim. We assume that a notification email (“Welcome . . .”) sent by the cloud storage service to the victim after step (1) will be ignored as spam by most users.

All providers have been informed about the problem.

## About the Fraunhofer Institute for Secure Information Technology

The Fraunhofer Institute for Secure Information Technology SIT is the leading expert for IT Security and develops solutions for immediate use, tailored to the customer's needs.

Over one hundred highly qualified employees covering all areas of IT security make such customized services possible. The staff constitutes the competency foundation for cross-technological services at the highest level.

The Fraunhofer Institute SIT is active in projects for companies from all kinds of industries. Numerous successful projects carried out with international partners are the resounding proof for trustful and reliable cooperation.

ISBN: 978-3-8396-0391-8