

# redbackup: a redundant distributed backup system prototype

Fabian Hauser  
Raphael Zimmermann

December 11, 2017

## 1 Client and Supervisor

**Supervisor:** Prof. Dr. Farhad Mehta, HSR Rapperswil

**Client:** IFS (Institut für Software), HSR

## 2 Students

- Raphael Zimmermann, rzimmerm@hsr.ch
- Fabian Hauser, fhauser@hsr.ch

## 3 Setting

Today, most individuals, as well as small and medium enterprises (SME), create backups with local devices (mostly external hard disks), which are not physically distributed<sup>1</sup>. Alternatively, some SME, and individuals store their data online using cloud storage providers, but legal issues, privacy concerns and the dependency on large providers limit the widespread use of cloud storage. Currently, the security of cloud backups is based solely on the strength of cryptographic algorithms and security measures taken by cloud providers.

Automated backup solutions, which store the data in a decentralised and distributed manner within a trusted network could solve this issue. Today, however, there is no ready to use solution available on the market.

## 4 Vision

A possible solution for a efficient, decentralised and distributed backup could be realised with a combination of two parts,

1. a distributed storage system, which provides redundant, deduplicated storage and
2. an easy to use client software, which creates encrypted backups and uploads the data into the distributed system.

To permit a widespread usage of such a solution, it is crucial that both parts are easy to install and configure.

---

<sup>1</sup>The issue of local backups was brought to broad public attention in may 2017 due to wide spread infections of the ransomware "WannaCry" that also encrypted files on attached devices and network shares. See [3, 2]

**The distributed system** as proposed above (1) should be able to:

- a. build a peer-to-peer network of trusted nodes over the internet.
- b. handle joining of nodes including authentication.
- c. handle planned and unplanned leaving of nodes.
- d. store data with a defined degree of redundancy.
- e. protect data from being modified or deleted<sup>2</sup>.  
The system may discard data after it has reached a specified expiration date.
- f. address and retrieve stored data.
- g. allow access from clients including authentication over an API.

**The client software** as proposed above (2) should be able to:

1. create backups of files and directories (content and metadata).
2. compress and encrypt backups.
3. upload backups into the distributed system (see 1).
4. retrieve, decrypt, decompress and recover data from the distributed system (see 1).

## 5 Goals

The goal of the Study Project is to provide a theoretical description of an append only, distributed peer-to-peer data storage as the basis for the described vision, as well as a working prototype.

**The theoretical concept** must address the following issues:

- a. joining of nodes
- b. planned and unplanned leaving of nodes
- c. distribution of data with a defined degree of redundancy
- d. fair distribution of data blocks within the distributed system
- e. uploading data into the distributed system
- f. addressing within the distributed system
- g. retrieving stored data
- h. scalability for up to several 100 nodes where every node can store a data volume of up to 2 terabytes.
- i. detection of inconsistent and unresolvable states (e.g. redundancy failures because there are not enough nodes)

All the above problems will be implemented in a prototype to demonstrate the described concepts. It is not intended for production usage. The concept, as well as the prototype, assumes that all nodes are addressable by a fixed IP-address and port. Encryption and deduplication of data on the client will not be discussed within this study project since it is limited to uploading and retrieval of data.

A short evaluation will be carried out at the beginning of the project in order to find an appropriate implementation language for the prototype. The current candidates for a suitable implementation language are Rust and Erlang.

---

<sup>2</sup>The idea of an append only approach prevents accidental or malicious deletion of stored data, for example by ransomware.

## 6 License

The study project is intended to be free software and will be published under the AGPL-License [1].

## 7 Guidelines

The students and the supervisor will plan weekly meetings to check and discuss progress.

All meetings are to be prepared by the students with an agenda. The agenda will be sent at least 24h prior to the meeting. The results will be documented in meeting minutes that will be sent to the supervisor.

A project plan must be developed at the beginning of the thesis to promote continuous and visible work progress. For every milestone defined in the project plan, the temporary versions of all artefacts need to be submitted. The students will receive a provisional feedback for the submitted milestone results. The definitive grading is however only based on the final results of the formally submitted report.

## 8 Documentation

The project must be documented according to the regulations of the Computer Science Department at HSR (see <https://www.hsr.ch/Allgemeine-Infos-Bachelor-und.4418.0.html>). All required documents are to be listed in the project plan. All documents must be continuously updated, and should document the project results in a consistent form upon final submission. The documentation has to be completely submitted in 3 copies on CD/DVD. A printed version has to be delivered if requested by the supervisor.

## 9 Important Dates

Refer to <https://www.hsr.ch/Termine-Diplom-Bachelor-und.5142.0.html>.

## 10 Workload

A successful study project results in 8 ECTS credit points per student. One ECTS points corresponds to a work effort of about 30 hours. All time spent on the project must be recorded and documented.

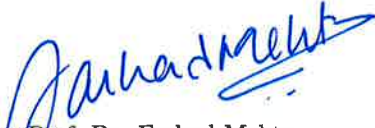
## 11 Grading

The HSR supervisor is responsible for grading the study project. The following table gives an overview of the weights used for grading.

Facet	Weight
1. Organisation, Execution	1/5
2. Report	1/5
3. Content	3/5

The effective regulations of the HSR and Department of Computer Science apply (see <https://www.hsr.ch/Ablaeufe-und-Regelungen-Studie.7479.0.html>).

Rapperswil, December 2017



Prof. Dr. Farhad Mehta  
12.12.2017

## References

- [1] Gnu affero general public license. <https://www.gnu.org/licenses/agpl-3.0.en.html>.
- [2] M. Scott and N. Wingfield. Hacking Attack has security experts scrambling to contain fallout. <https://www.nytimes.com/2017/05/13/world/asia/cyberattacks-online-security-.html>, 2017.
- [3] J. C. Wong and O. Solon. Massive ransomware cyber-attack hits nearly 100 countries around the world. <https://www.theguardian.com/technology/2017/may/12/global-cyber-attack-ransomware-nsa-uk-nhs>, 2017.