

# Thesis Proposals

Here we gather some ideas for theses that could be written by students. All abstracts show two phases, e.g. studying possible options for enhancements and then coding them. Feel free to [contact us](#) if one of the topics meets your interests or field of studies.

## Delta Backups

For some workloads it is useful to backup the binary differences between two files and as such the changed blocks only. Therefore it makes sense to look into the different fingerprinting techniques that exist nowadays. This thesis work is mostly an investigation into those different techniques and determine what OSS libraries could be used for this, as for example librsync, rabin fingerprinting etc.

The thesis work therefore consists of two parts: first examining the current available libraries to be able to evaluate them in terms of adding benefits to the backup-process in certain workload-scenarios and also describing those scenarios.

The second part is coding in C/C++. As this work would function as an extension to the so called File Daemon of the BAREOS system and we have added a Python Plugin interface it might be possible to create a prototype in Python.

## Crypto Framework

Currently we mostly use OpenSSL as our primary crypto framework. Recently we added abstraction to allow different crypto frameworks. We would like to extend this functionality by doing some thesis work.

This thesis work consists of investigating how the different crypto frameworks are interfaced through their API and then implement at least one alternative crypto framework using the abstraction. Currently we support TLS OpenSSL and GNUTLS, but for crypto we support OpenSSL only. We have some dummy prototypes for Mozilla NSS which we want to use for TLS and Crypto eventually. The thesis work also involves investigating any other OSS crypto frameworks that might be used.

So the thesis work consists of two parts: first examining the current available frameworks and evaluate them as to how they could be used as an alternative to OpenSSL. The second part would be coding in C/C++.

## Network Connectivity

The current Bareos daemons all transfer data using the TCP protocol. For both high latency and high distance connections this may not scale too well though, as TCP has known scaling problems.

An alternative could lie within the UDT protocol, which addresses most of these problems. Over time even more substitute protocols have been added to most OSes, which could be suitable to replace TCP (e.g. SCTP as a close to one-to-one replacement).

In the course of this thesis work, as a first step we would like to investigate the different new protocols that are available and get to know their pros and cons. We have already done some preliminary work to abstract our socket connections, so the second step would then be to implement promising protocols and possibly change the abstraction to fit them, thereby ensuring full functionality of the old code. The coding is to be done in C/C++.

For additional Information see <http://udt.sourceforge.net/>