Peer To Peer Systems & Security

Anonymous P2P VoIP System

Initial Approach Report

DHT - Subproject

Team Alpha

Elias Hazboun

Nedko Nedko

17.05.2015

The following is a report for the anonymous P2P VoIP Project – DHT subproject – undertaken by team Alpha which consists of the following team members:

- Elias Hazboun – elias.hazboun@tum.de
- Nedko Nedkov – nedko.nedkov@tum.de

Programming Language and OS:

The project shall be completed mainly using Python programming language, because we envision that DHT implementation is not processing heavy and thus will not be affected by the interpretation overhead of the language. Python is also easy to read/write and subsequently modify, so in that sense it is developer-friendly and we save time focusing on the functionality rather than language details. Python usually also requires less lines of code and with less lines of code less bugs are expected. Finally, the availability of open-source libraries is a big plus for anyone trying to develop networking/encryption programs. We will use Linux for the development, testing and deployment of the project; linux is networking and development friendly.

Build System:

Since our project will be written in Python, no build system is necessary. However, we shall write a Makefile that will for example check the project directory for all needed files and that all files are syntax-error free, using the file we would also be able to run unit-tests or simply clean the directory and get a fresh copy from the repository.

Quality Control:

We will use the python Unit testing framework. It is the de-facto standard unit testing framework for Python. Using it we can do test automation and aggregate tests into collections. The framework is easy to use since it provides classes that help developers in the testing.

For quality control, we shall use PyChecker – an open source static analysis tool which can help us in finding bugs in the source code.

For quality assurance, we will use Kwalitee – an open source tool for checking quality of the code in terms of formatting, syntax and indentation.

Libraries:

For the meantime, we cannot think of any 3rd party libraries that will be used, since we are still in the early phases of the project and the default libraries provided in Python have a great functionality covered.

License:

Our project shall be available in the public domain as an open source project. We would like to contribute to the community and allow other developers to modify to their liking and enhance our original offering. MIT License will be used for our project; it is one the less restrictive licenses for software. It is very famous and simple, anybody is allowed to do anything with our code, as long as they include in their project a statement that testifies to the existence of original code by us and include the original license

Team Experience:

We both have previous experiences with low level socket programming and have previously done networking projects in Python on linux. Elias, having taken Advanced Computer network last semester with this chair, has experience in Mininet usage to virtualize peers in the network which will be used to test our DHT implementation.

Workload Sharing:

We will separate the workload into four major tasks:

- Design: Load shall be split as evenly as possible between us.
- Implementation: Load shall be split as evenly as possible between us.
- Testing: Load shall be split 70-30, with more work on Nedko.
- Documenting: Load shall be split 70-30 with more work on Elias.

More granular task sharing is done at a later stage in the project.

Issues and Complaints:

None.