582615 - Overlay and P2P Networks, Spring 2016

Exercise package 2 (20 points)

The exercises are intended to be done working in pairs. This package contains four exercises and an optional turbo challenge. During the course there will be three sets of exercises. The course book and the lectures contain some answers, but searching for outside sources too is encouraged.

Return your answers by email to juhani.toivonen@cs.helsinki.fi as an attached PDF or TXT document. Use “Overlay Exercise 2” as the subject line. The document should include:

- The title “Overlay exercise package 2”
- The name and student number of both writers
- The answers to the exercises

There are two types of exercise sessions: Clarification sessions, where you can ask questions about the exercises or other matters about the course; and Answer sessions, where some answers to the exercises are presented and discussed.

- Clarification session: Wednesday 3.2. at 12:15
- **Exercise deadline**: Tuesday 9.2. at 23:59
- Answer session: Wednesday 10.2. at 12:15

Assignments

Assignment 1 - BitTorrent (5 points)

BitTorrent relies on two key algorithms: *Choke algorithm* for peer selection, and *Rarest first algorithm* for piece download strategy. Describe:

- What does the *Rarest first algorithm* do?
- What do the three added policies do? When are they used and what is their goal?
- What does the *Choke algorithm* do? How are new joining peers with nothing to give taken into account?
- Starting from mainline BitTorrent 4.0.0, clients that have finished downloading and decide to stay seeding the torrent use a modified version of the *Choke algorithm*. How is this modified version different?

You can find descriptions about the algorithms in *“Rarest first and choke algorithms are enough”* published by Arnaud Legout, G. Urvoy-Keller and P. Michiardi at ACM SIGCOMM ’2006. (http://conferences.sigcomm.org/imc/2006/papers/p20-legout.pdf).

Assignment 2 - Bloom Filters (5 points)

A Bloom filter (named after its inventor Burton Howard Bloom) is a deliberately non-error-free data-structure. Concentrate on the standard Bloom filter and describe:

- What are Bloom filters and what are they used for?
- What is the difference between a Bloom filter and a regular hash table?
- Describe the steps of inserting an item into a Bloom filter, and querying the presence of that item.
- Can elements be removed from a standard Bloom filter? Explain why.
- Bloom filters sometimes return false results. Doesn’t that make them useless? What *do* we know about the results?

Assignment 3 - Distributed Hash Tables (DHTs) (5 points)

Distributed Hash Tables, like hash tables, are systems for storing (key, value) pairs, but in a setting that can span multiple hosts. Describe:

- What are typical operations that a DHT provides?
- What challenges does spreading a hash table over multiple nodes impose?
- Choose a DHT (e.g., Kademlia, CAN, Tapestry or Chord) and describe the steps of inserting a (key, value) pair into that DHT. What can you say in general about retrieving that value from the DHT?

Assignment 4 - Consistent hashing (5 points)

Consistent hashing is a technique used for load balancing and minimizing the effort of redistributing keys when changing the amount of nodes in a DHT. Describe:

- How does consistent hashing work?
- What happens when you add/remove a node in a system that uses consistent hashing?
- Why is it important to use a well balanced hash function?
- How can replication be done with consistent hashing?

Turbo challenge (5 points)

The turbo challenge allows you to recover lost points from other assignments, but will not increase the maximum points available. You can get full points from the exercise set without the turbo challenge. The turbo challenge points will carry over to other exercise sets.

Cassandra is a high performance distributed data store that is used in the back-end of several popular Internet services worldwide.

- How does Cassandra partition the data?
- What does Cassandra do when it notices that the data is distributed unequally among the nodes?
- What techniques does Cassandra use achieve scalability and redundancy?

References:

- http://cassandra.apache.org
- http://www.cs.cornell.edu/projects/ladis2009/papers/lakshman-ladis2009.pdf