Test case: Log processing

Task

Convert a set of log files (input files) to a more efficient set of log files (output files).
Download the input files here:

https://docs.google.com/open?id=0ByFmJZ6CrEfUWDd1bjZQS29nQUk

The output files should contain:

- timestamp
- IP
- Device type (Pc, Mobile, Tablet, Tv)
- Compressed user agent

You will see just a few distinct URLs in the logs. Only process lines with this URL: http://scripts.adrcdn.com/000394/scripts/screenad_launch_9.4.0_scrambled.js

Goals

- The processing should be as fast and efficient as possible, making optimal use of memory, CPUs and disk performance
- Try compressing the user agent to a far smaller bytesize. A very small chance of collisions (two agents compressing to the same bytes) is acceptable.
- Try to code so that when you restart a job after a failure or interruption, it picks up roughly where it stopped
- Dynamically get the list of log files (not hard coded in the source)

How

- You can choose your own tech stack (language, DBs, OS, etc)
- We encourage you to use any existing libraries or executables to handle subtasks

Deliverables

- Your source code
- A benchmark (e.g. logs processed in X seconds on a 2Ghz i5 with 4GB mem)
- Notes about your code and choices

Also

It’s a task that enables us to see your approach in dealing with different stacks and coding techniques. However, it will probably never go into production, and therefore doesn’t have to be foolproof. Also, spend only the time you feel comfortable with.