Design a load-balancer

Evaluation case for node.js developers

The load-balancer shall balance requests from clients between the video-servers case1-1.neti.systems, case1-2.neti.systems and case1-3.neti.systems. The video-servers listen on port 3000. These (fake) video-servers shall be up and running now and you can make requests to them.

The requests from the clients are http POST requests to /allocateStream with a json body containing a channel identifier. A client request to the load-balancer can be simulated with

```
curl -H "Content-Type: application/json" -X POST -d '{"channelId":"svt1"}' \
  "http://localhost:3000/allocateStream"
```

The load-balancer shall balance the requests in a round-robin fashion between the three available video-servers. The requests from the load-balancer to the video-servers are identical to the request from the client to the load-balancer. The video-server responds to the request with a small json-structure. The load-balancer shall remove the property "secret" from the response and return it to the client. A typical response from a video-server can be

```
{
    "url": "http://video1.neti.systems/svt1?token=12345",
    "secret": "abcdef"
}
```

The load-balancer shall then return the following response to the client:

```
{
    "url": "http://video1.neti.systems/svt1?token=12345"
}
```

Sometimes, the video-servers respond with a 500 error code. In that case, the load-balancer shall try with another video-server instead.

Sometimes, the video-servers are slow to respond. If the load-balancer has not received a response within 1 second, it shall try with another video-server instead.

If all video-servers respond with 500 or fail to respond, the load-balancer shall return a 500 error to the client.

Solution requirements

Your solution shall be written in node.js and it shall be possible to run it on either node 4.2.6 or node 5.5.0. You are encouraged to use modules on npm as part of your solution.

Your solution shall also include tests for your own code.

Please deliver your solution in a .zip or .tar.gz-file that we can extract and run your code to test that it works.