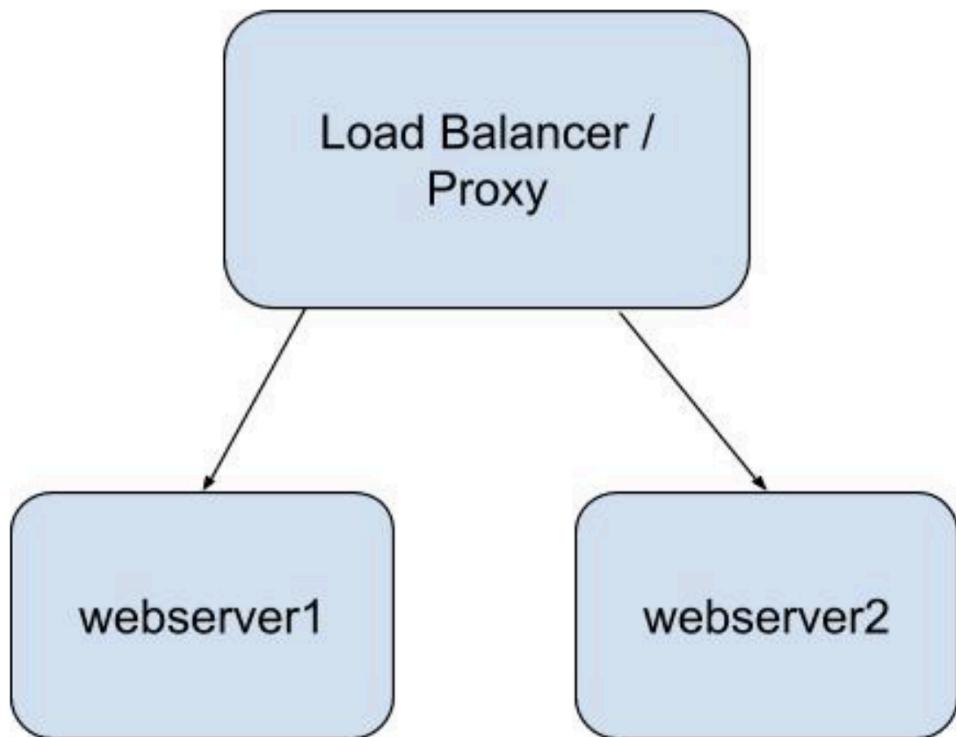


**Exercise-1:**

Build a load-balanced web server environment



**Load Balancer / Proxy:**

You are free to pick another AWS/GCP load balancer

**Web server:**

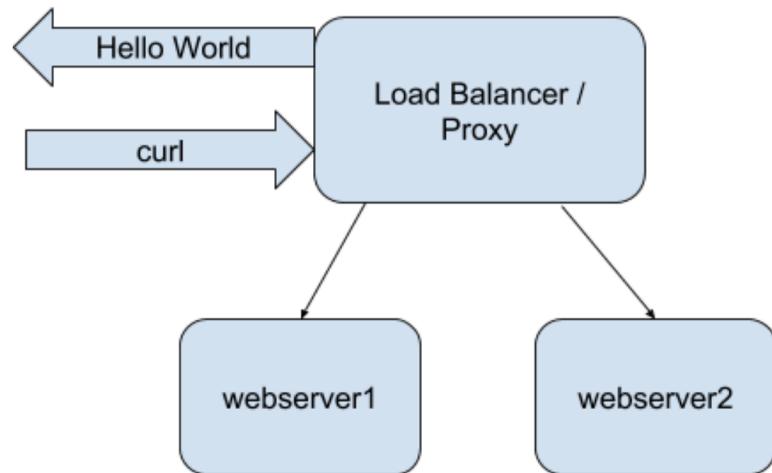
We recommend using nginx. You are free to pick another web server as well.

**Expected Results:**

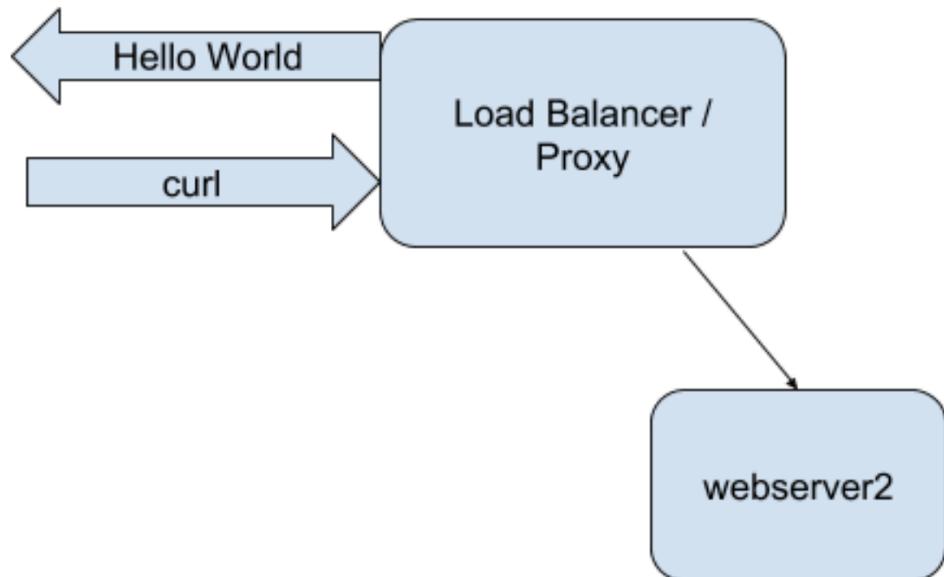
Curl of your Load Balancer should return a “Hello World” that is being served by one of your web servers.

Removal of either one of your web servers should automatically failover to the other server

Both Web Servers Enabled



One Web Server Removed



### **Guidelines:**

Please do this exercise in AWS/GCP with your own lab account. Selection of OS, config management, and software packages are up to you. Follow the guidelines for the required tools.

### **Required Tools:**

- Terraform - To build the infrastructures, install the webservers and create the load balancers. Try using terraform to the utmost to build it out
- FOSS (Free and open-source software)
- External software dependencies must be hosted on the public internet

### **Submission:**

- Do not include binaries
- Deadline: 1 week from the time you received this document
- Share your Github/gitlab/bitbucket repo for the code you develop.
- Readme file on how to re-create your environment

### **Review Process:**

- We will sit with the candidate to review the code and they need to explain how things have been done

### **Bonus:**

- Brief summary of what you liked about your solution
- Brief summary of what you disliked about your solution

### **Exercise-2:**

Use any programming languages(**Preferably Python**) to solve the any one of the below use cases:

1. Any AWS services that are created in an account should automatically be tagged using the user ARN who is creating it out.  
(or)
2. AWS user access provision should be done via a programming language.

### **Review Process:**

- We will sit with the candidate to review the code and they need to explain how things have been done

### **Bonus:**

- Brief summary about how the use case can be scalable.
- Brief summary of what you disliked about your solution.