In this first task you need to configure web-02 to be identical to web-01. Fortunately, you built a Bash seriet during your web server project (/rltoken/-JluPVwfvXMOYMzNOqvgsQ), and they'll now come in handy to easily configure web-02. Remember, always try to automate your work!

Since we're placing our web servers behind a load balancer for this project, we want to add a custom Nginx response header. The goal here is to be able to track which web server is answering our HTTP requests, to understand and track the way a load balancer works. More in the coming tasks.

Requirements:

- Configure Nginx so that its HTTP response contains a custom header (on web-01 and web-02)
 - The name of the custom HTTP header must be X-Served-By
 - The value of the custom HTTP header must be the hostname of the server Nginx is running on
- Write 0-custom_http_response_header so that it configures a brand new Ubuntu machine to the requirements asked in this task
 - Ignore (/rltoken/k3Bt6zu1On_-mDszxi0Z9w) SC2154 (/rltoken/9KwKHb9H8OJqcSK0saRIOA)
 for shellcheck

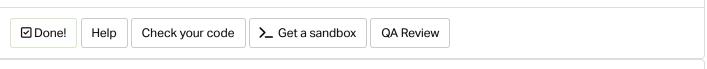
Example:

```
sylvain@ubuntu$ curl -sI 34.198.248.145 | grep X-Served-By
X-Served-By: 03-web-01
sylvain@ubuntu$ curl -sI 54.89.38.100 | grep X-Served-By
X-Served-By: 03-web-02
sylvain@ubuntu$
```

If your server's hostnames are not properly configured ([STUDENT_ID]-web-01 and [STUDENT_ID]-web-02.), follow this tutorial (/rltoken/qSor8ulAHI4HedrO6KJEoQ).

Repo:

- GitHub repository: alx-system_engineering-devops
- Directory: 0x0F-load_balancer
- File: 0-custom_http_response_header



1. Install your load balancer

mandatory

Score: 0.0% (Checks completed: 0.0%)

Install and configure HAproxy on your lb-01 server.

Requirements:

- Configure HAproxy so that it send traffic to web-01 and web-02
- Distribute requests using a roundrobin algorithm
- Make sure that HAproxy can be managed via an init script

- Make sure that your servers are configured with the right hostnames: [STUDENT_ID]-web-01 and [STUDENT_ID]-web-02. If not, follow this tutorial (/rltoken/YkfzgEa6xNHrQbkKmJN4zg).
- For your answer file, write a Bash script that configures a new Ubuntu machine to respect above requirements

Example:

sylvain@ubuntu\$ curl -Is 54.210.47.110

HTTP/1.1 200 OK

Server: nginx/1.4.6 (Ubuntu)

Date: Mon, 27 Feb 2017 06:12:17 GMT

Content-Type: text/html

Content-Length: 30

Last-Modified: Tue, 21 Feb 2017 07:21:32 GMT

Connection: keep-alive ETag: "58abea7c-1e" X-Served-By: 03-web-01 Accept-Ranges: bytes

sylvain@ubuntu\$ curl -Is 54.210.47.110

HTTP/1.1 200 OK

Server: nginx/1.4.6 (Ubuntu)

Date: Mon, 27 Feb 2017 06:12:19 GMT

Content-Type: text/html
Content-Length: 612

Last-Modified: Tue, 04 Mar 2014 11:46:45 GMT

Connection: keep-alive ETag: "5315bd25-264" X-Served-By: 03-web-02 Accept-Ranges: bytes

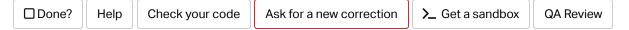
sylvain@ubuntu\$

Repo:

• GitHub repository: alx-system_engineering-devops

• Directory: 0x0F-load_balancer

• File: 1-install_load_balancer



2. Add a custom HTTP header with Puppet

#advanced

Score: 0.0% (Checks completed: 0.0%)



Just as in task #0, we'd like you to automate the task of creating a custom HTTP header response, but with Puppet.

- The name of the custom HTTP header must be X-Served-By
- (/). The value of the custom HTTP header must be the hostname of the server Nginx is running on
 - Write 2-puppet_custom_http_response_header.pp so that it configures a brand new Ubuntu machine to the requirements asked in this task

Repo:

- GitHub repository: alx-system_engineering-devops
- Directory: 0x0F-load_balancer
- File: 2-puppet_custom_http_response_header.pp

Done? Help Check your code	Ask for a new correction	>_ Get a sandbox	QA Review
----------------------------	--------------------------	------------------	-----------

Copyright © 2024 ALX, All rights reserved.

