Chapter 6

Lab 6.1

Introduction

In this lab we will create an instance of Dromedary, and we will add it to an HAProxy load balancer. Then, we will create another instance of Dromedary on a separate port, and add it to the load balance pool. Lastly, we will remove the old instance of Dromedary from the load balance pool to demonstrate **blue/green** deployments.

Install the first instance of Dromedary

1. Log into your Vagrant virtual machine via **vagrant up** or **vagrant ssh** if it is already running.

2. Ensure Git is installed:

   $ sudo apt-get install -y git

3. Ensure Java 8 is installed:

   $ sudo add-apt-repository ppa:openjdk-r/ppa

   followed by

   $ sudo apt-get update

   and, lastly,

   $ sudo apt-get install -y openjdk-8-jdk

4. Install **node.js** and **npm**:

   $ sudo apt-get install nodejs npm

5. Symlink **nodejs** to **node**:

   $ sudo ln -s /usr/bin/nodejs /usr/bin/node

6. Afterwards, clone the Dromedary repository via:

```
$ git clone https://github.com/liatrio/dromedary.git
```

7. Change the directory to **dromedary** via:

```
$ cd dromedary
```

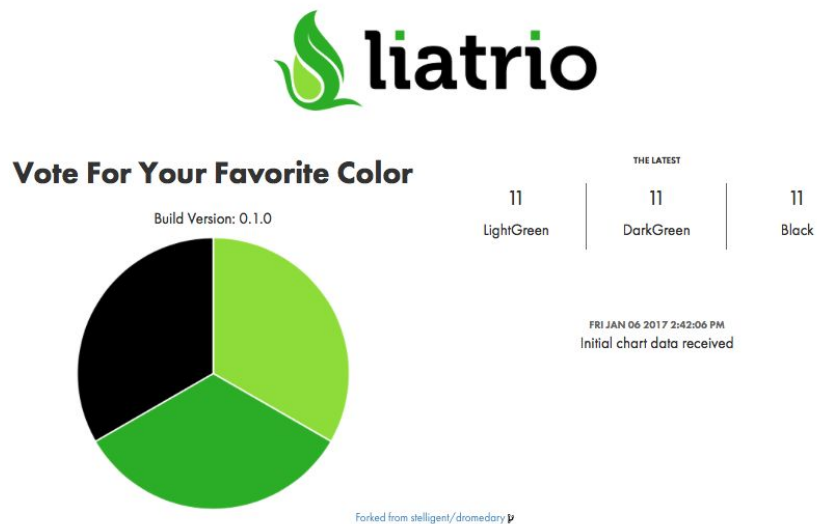8. Install the node dependencies:

```
$ npm install
```

9. Run the following command to install the **gulp** tool:

```
$ sudo npm install -g gulp
```

10. Start the first instance of Dromedary, and fork the process to the background via **PORT=8080 nohup gulp &.**

11. Verify the application is running by browsing to **http://localhost:18080/**. You should see the Dromedary application running.

Install and configure HAProxy

1. Install **haproxy**:

```
$ sudo apt-get install haproxy
```

2. Rename the original **haproxy.cfg** file:

```
$ sudo mv /etc/haproxy/haproxy.cfg /etc/haproxy/haproxy.cfg.orig
```

3. Create a new **haproxy** configuration with the following contents:

```
$ sudo vi /etc/haproxy/haproxy.cfg

global
        log /dev/log    local0
        log 127.0.0.1   local1 notice
        maxconn 4096
        user haproxy
        group haproxy
        daemon
defaults
        log     global
        mode    http
        option  httplog
        option  dontlognull
        retries 3
        option redispatch
        maxconn 2000
        contimeout      5000
        clitimeout      50000
        srvtimeout      50000
listen webfarm 0.0.0.0:80
    mode http
    stats enable
    stats uri /haproxy?stats
    balance roundrobin
    option httpclose
    option forwardfor
    server webserver01 localhost:8080 check
```

4. Ensure that the default start script is enabled for **haproxy**:

```
$ sudo vi /etc/default/haproxy
```

and uncomment the **ENABLED** line:

```
# Set ENABLED to 1 if you want the init script to start haproxy.
ENABLED=1
```

5. Add extra flags here:

```
# EXTRAOPTS="-de -m 16"
```

6. Next, restart the **haproxy** service with the new configuration:

```
$ sudo service haproxy restart
```

7. Verify **haproxy is** load balancing by browsing to **http://localhost:1080/**.

8. Review the **haproxy** status by browsing to **http://localhost:1080/haproxy?stats**.

HAProxy version 1.4.24, released 2013/06/17

Statistics Report for pid 5439

> **General process information**

pid = 5439 (process #1, nbproc = 1)
uptime = 0d 0h00m14s
system limits: memmax = unlimited; ulimit-n = 8204
maxsock = 8204; maxconn = 4096; maxpipes = 0
current conns = 1; current pipes = 0/0
Running tasks: 1/2

active UP
active UP, going down
active DOWN, going up
active or backup DOWN
active or backup DOWN for maintenance (MAINT)
Note: UP with load-balancing disabled is reported as "NOLB".

backup UP
backup UP, going down
backup DOWN, going up
not checked

Display option:
- Hide 'DOWN' servers
- Refresh now
- CSV export

External resources:
- Primary site
- Updates (v1.4)
- Online manual

webfarm

	Queue			Session rate			Sessions					Bytes		Denied		Errors				Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				1	8	-	1	2	2 000	12		11 002	45 516	0	0	0					OPEN								
webserver01	0	0	-	0	8		0	2	-	10	10	10 013	37 412		0		0	0	0	0	14s UP	L4OK in 0ms	1	Y	-	0	0	0s	-
Backend	0	0		0	8		0	2	2 000	10	10	11 002	45 516	0	0		0	0	0	0	14s UP		1	1	0		0	0s	

Create the second instance of Dromedary

1. Ensure the present working directory is the base directory of the **dromedary** repository.

2. Start the second instance of Dromedary and fork the process to the background via **PORT=8081 nohup gulp &**.

3. Add the second instance to the **haproxy** load balance pool by editing **/etc/haproxy/haproxy.cfg** and appending at the end of the file:

```
$ server webserver02 localhost:8081 check
```

4. Reload the **haproxy** service for the changes to take effect:

```
$ sudo service haproxy reload
```

5. Verify both nodes are in the load balance pool by browsing to **http://localhost:1080/haproxy?stats** as well as **http://localhost:1080/**.

THE LINUX FOUNDATION

HAProxy version 1.4.24, released 2013/06/17

Statistics Report for pid 1928

> General process information

pid = 1928 (process #1, nbproc = 1)
uptime = 0d 0h00m10s
system limits: memmax = unlimited; ulimit-n = 8205
maxsock = 8205; maxconn = 4096; maxpipes = 0
current conns = 1; current pipes = 0/0
Running tasks: 1/3

active UP / active UP, going down / active DOWN, going up / active or backup DOWN / active or backup DOWN for maintenance (MAINT)
backup UP / backup UP, going down / backup DOWN, going up / not checked
Note: UP with load-balancing disabled is reported as "NOLB".

Display option:
- Hide 'DOWN' servers
- Refresh now
- CSV export

External resources:
- Primary site
- Updates (v1.4)
- Online manual

webfarm

	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings				Server						
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				1	5	-	1	2	2 000	11		10 425	13 949	0	0	0					OPEN								
webserver01	0	0	-	0	3	-	0	1	-	5	5	5 190	4 295		0		0	0	0	0	10s UP	L4OK in 0ms	1	Y	-	0	0	0s	-
webserver02	0	0	-	0	2	-	0	1	-	4	4	4 246	830		0		0	0	0	0	10s UP	L4OK in 0ms	1	Y	-	0	0	0s	-
Backend	0	0		0	5		0	2	2 000	9	9	10 425	13 949	0	0		0	0	0	0	10s UP		2	2	0		0	0s	

Remove the old instance

1. Open **/etc/haproxy/haproxy.cfg** with a preferred text editor.

2. Remove the line: **server webserver01 localhost:8080 check.**

3. Reload the **haproxy** service for the changes to take effect:

   ```
   $ sudo service haproxy reload
   ```

4. Verify the node is removed by browsing to **http://localhost:1080/haproxy?stats** as well as **http://localhost:1080/**.

HAProxy version 1.4.24, released 2013/06/17

Statistics Report for pid 1971

> General process information

pid = 1971 (process #1, nbproc = 1)
uptime = 0d 0h00m02s
system limits: memmax = unlimited; ulimit-n = 8204
maxsock = 8204; maxconn = 4096; maxpipes = 0
current conns = 1; current pipes = 0/0
Running tasks: 1/2

active UP / active UP, going down / active DOWN, going up / active or backup DOWN / active or backup DOWN for maintenance (MAINT)
backup UP / backup UP, going down / backup DOWN, going up / not checked
Note: UP with load-balancing disabled is reported as "NOLB".

Display option:
- Hide 'DOWN' servers
- Refresh now
- CSV export

External resources:
- Primary site
- Updates (v1.4)
- Online manual

webfarm

	Queue			Session rate			Sessions					Bytes		Denied		Errors			Warnings				Server						
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				1	1	-	1	1	2 000	1		0	0	0	0	0					OPEN								
webserver02	0	0	-	0	0	-	0	0	-	0	0	0	0		0		0	0	0	0	2s UP	L4OK in 0ms	1	Y	-	0	0	0s	-
Backend	0	0		0	0		0	0	2 000	0	0	0	0	0	0		0	0	0	0	2s UP		1	1	0		0	0s	

THE LINUX FOUNDATION