==========

HAProxy and Nginx Playbook Reference Manual v1.0 Rough Draft  Prabhakaran Varadarajan

==========

Environment: Production

Ansible version: 2.3.1.0

OS Version: CentOS 6.X

Cloud platform: AWS

HAProxy Version: 1.5.18

Nginx Version: 1.12.1

Selinux mode: enforcing

Please refer host information for ansible in hosts file which located in /etc/ansible/hosts.
There are two groups created "lbservers" and "webservers" respectively.

Test1 - 192.168.x.x [lbservers]

Test2 - 192.168.x.x [webservers]

Test3 - 192.168.x.x [webservers]

Deploy the site by,

ansible-playbook -i hosts site.yml

site.yml file which contains aforementioned groups with roles which will execute the specified tasks. Once it is executed, haproxy will be installed/configured. Loadbalance service will listen on port 80. Nginx installation and hosting will also take place. Nginx web service listens on port 8080 on both the nodes Test2 and Test3. Both are part of webpool backend[app] which sits behind loadbalancer and handles actual web requests.

Frontend HAProxy receives web traffic on port 80 and forward it to webpool on port 8080 for processing.

Loadbalance algorithm(roundrobin) used with upstream checks along with "fall" and "rise" as response validation.

Script "prepareURLredir.sh" located in "/etc/ansible/roles/webservers/files" has been setup to do preparatory work such as directory/file creation and setting up permission for web service.

Virtual host configured with configuration file name as "www.webexample.com.conf" with Directories /probe_local as well as /probe_listing. /probe_local will reflect contents of /var/www/local.html.

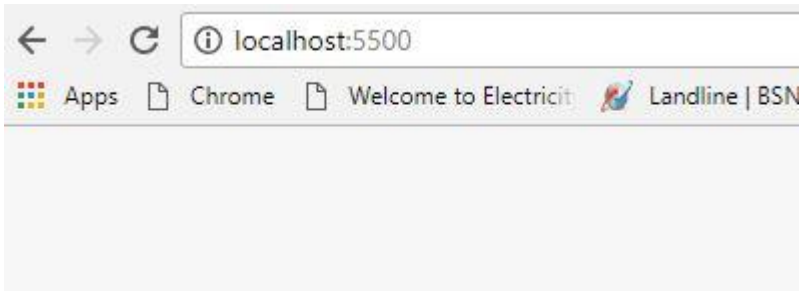
"/probe_listing" - setup made for directory listing(autoindex used here).

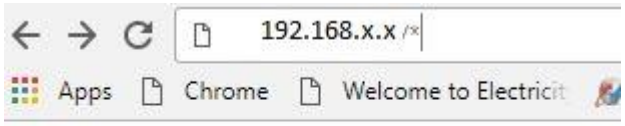
"/*" - forward to www.redhat.com site. This has been accomplished using rewrite rule.

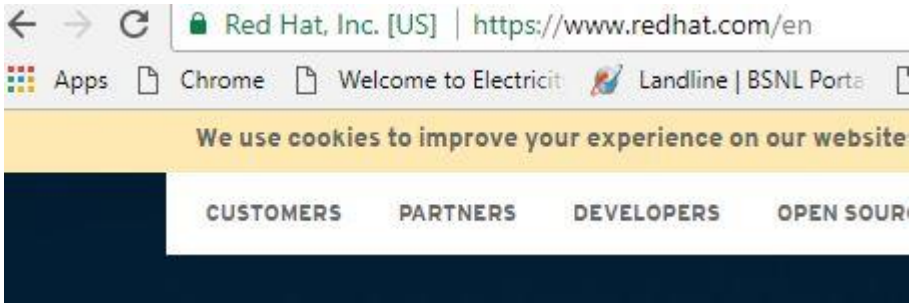
"/probe_remote" - forward to localhost:5500(for local application) This has been accomplished using rewrite rule.

Try the URL access by,

http://Test1/probe_local/


Production ready Configuration and Tuning

==========

HAproxy production settings maintained by removing "stats" URL etc.,

For Nginx, removed default settings and configured virtual host with directories. Sysctl parameter "net.core.somaxconn" set to 4096 to handle more number of requests. "backlog parameter also added" in the template.

Facts variable "ansible_processor_count" used to taking into account when it comes to connection volumes. Alternate option is to set "auto".

Reboot safe

==========

chkconfig enabled for haproxy, nginx for this purpose.

Selinux

=======

You may be getting "403-Forbidden Error" when you enforcing selinux mode. Fix it by,

chcon -R -t httpd_sys_content_t /var/www

!!!!!!!!!!!!!!!!!!!!!

Code

[root@ip-192-168-30-11 ansible]# pwd

/etc/ansible

[root@ip-192-168-30-11 ansible]# tree

.

├──ansible.cfg

├──group_vars

│  ├──lbservers.yml

│  └──webservers.yml

├──hosts

├──host_vars

├──production.ini

├──README.md

├──roles

│  ├──lbservers

│  │  ├──handlers

```
│   │   │   └── main.yml
│   │   ├── tasks
│   │   │   └── main.yml
│   │   └── templates
│   │       └── haproxy.cfg.j2
│   └── webservers
│       ├── files
│       │   └── prepareURLredir.sh
│       ├── handlers
│       │   └── main.yml
│       ├── tasks
│       │   └── main.yml
│       └── templates
│           ├── nginx.conf.j2
│           └── www.webexample.com.conf.j2
├── site.retry
└── site.yml

12 directories, 16 files
[root@ip-192-168-30-11 ansible]#
```

```
[root@ip-192-168-30-11 ansible]# cat site.yml
---

- name: Main HAProxy [LoadBalance] configuration and start/restart
  hosts: lbservers
  remote_user: ansible
  become: yes
  become_method: sudo

  roles:
    - lbservers
```

```
- name: Main configuration for Nginx [Hosting]
    hosts: webservers
    remote_user: ansible
    become: yes
    become_method: sudo

    roles:
       - webservers

[root@ip-192-168-30-11 ansible]#

[root@ip-192-168-30-11 ansible]# cat production.ini
# Production Environment File

[lbservers]
Test1

[webservers]
Test2
Test3

[all:vars]
# environment for production
environment=prod
[root@ip-192-168-30-11 ansible]# cat hosts
## Hosts File Entries for Ansible Playbook

##LB Server
[lbservers]
Test1 ansible_ssh_host=192.168.30.11 ansible_ssh_user=ansible
```

##Web Server

[webservers]

Test2 ansible_ssh_host=192.168.30.12 ansible_ssh_user=ansible

Test3 ansible_ssh_host=192.168.30.13 ansible_ssh_user=ansible

[root@ip-192-168-30-11 ansible]#

[root@ip-192-168-30-11 group_vars]# cat lbservers.yml webservers.yml

---

# Variables for HAProxy

# HAProxy Listening port configured as 80 as per requirement, mode as http and LB algorithm as roundrobin

haproxy_port: 80

#HAProxy http support mode

haproxy_mode: http

# LoadBalancing Algorithm

haproxy_algorithm: roundrobin

---

# Nginx site specific config file variables

# Nginx Listen on Port 8080

nginx_listening_port : 8080

# BackLog Parameter Value - Nginx Tuning, part of somaxconnection kernel parameter

blparam : 4096

[root@ip-192-168-30-11 group_vars]#

[root@ip-192-168-30-11 handlers]# pwd

/etc/ansible/roles/lbservers/handlers

[root@ip-192-168-30-11 handlers]# cat main.yml

---

```
# Handlers for HAproxy

- name: restart haproxy
    service: name=haproxy state=restarted

- name: reload haproxy
    service: name=haproxy state=reloaded
[root@ip-192-168-30-11 handlers]#

[root@ip-192-168-30-11 templates]# cat haproxy.cfg.j2
global
        #local2.*       /var/log/haproxy.log
        log             127.0.0.1 local2
        chroot          /var/lib/haproxy
        pidfile         /var/run/haproxy.pid
        maxconn         4000
        user            haproxy
        group           haproxy
        daemon

        # turn on stats unix socket
        stats socket /var/lib/haproxy/stats

defaults
        mode                    http
        log                     global
        option                  httplog
        option                  dontlognull
        option                  http-server-close
        option forwardfor       except 127.0.0.0/8
        option                  redispatch
        retries                 3
```

```
        timeout http-request        10s
        timeout queue                1m
        timeout connect              10s
        timeout client               1m
        timeout server               1m
        timeout http-keep-alive 10s
        timeout check                10s
        maxconn                      3000

frontend    main
    bind *:{{ haproxy_port }}
    option http-server-close
    option forwardfor
    default_backend app

#---------------------------------------------------------------------
# round robin balancing between the backend
#---------------------------------------------------------------------
backend app
        mode {{ haproxy_mode }}
        balance       {{ haproxy_algorithm }}
        reqrep ^([^\ ]*\ /)probe_local[/]?(.*)          \1\2
        server    app1 192.168.30.12:8080 check fall 3 rise 2
        server    app2 192.168.30.13:8080 check fall 3 rise 2
[root@ip-192-168-30-11 templates]#

[root@ip-192-168-30-11 tasks]# pwd
/etc/ansible/roles/lbservers/tasks
[root@ip-192-168-30-11 tasks]# cat main.yml
---
# Install and Configuration
```

```
- name: Pull Package through Yum and install haproxy
    yum: name=haproxy state=present

- name: Configure HAProxy
    template: src=haproxy.cfg.j2 dest=/etc/haproxy/haproxy.cfg
    notify: restart haproxy

- name: Start/Restart the HAProxy service
    service: name=haproxy state=started enabled=yes
[root@ip-192-168-30-11 tasks]#

[root@ip-192-168-30-11 webservers]# cd files/
[root@ip-192-168-30-11 files]# ls
prepareURLredir.sh
[root@ip-192-168-30-11 files]# cat prepareURLredir.sh
#!/bin/bash

#Preparation for URL redirection

if [ ! -d /var/www/probe_listing ]; then
###echo "Directory not exists, so create it ..."
/bin/mkdir -p /var/www/probe_listing
fi

if [ ! -f /var/www/probe_listing/OncallRoster.csv ]; then
###echo "File not exists, so create it ..."
/bin/touch /var/www/probe_listing/OncallRoster.csv
fi

if [ "$HOSTNAME" == "ip-192-168-30-12.ap-south-1.compute.internal" ]; then
        echo "Test2" > /var/www/local.html
fi
```

```
if [ "$HOSTNAME" == "ip-192-168-30-13.ap-south-1.compute.internal" ]; then
        echo "Test3" > /var/www/local.html
fi

/bin/chown -R nginx:nginx /var/www/probe_listing
/bin/chown nginx:nginx /var/www/local.html

exit 0

[root@ip-192-168-30-11 files]#

[root@ip-192-168-30-11 tasks]# pwd
/etc/ansible/roles/webservers/tasks
[root@ip-192-168-30-11 tasks]# cat main.yml
---
- name: Add repository
  yum_repository:
    name: nginx
    description: NGINX YUM repo
    baseurl: http://nginx.org/packages/centos/6/x86_64/
    gpgcheck: no
    enabled: yes

- script: /etc/ansible/roles/webservers/files/prepareURLredir.sh

- sysctl:
    name: net.core.somaxconn
    value: 4096
    sysctl_set: yes
    state: present
    reload: no
```

```
- name: Install nginx
    yum: name=nginx state=present

- name: Apply nginx Site Specific Template
    template: src=www.webexample.com.conf.j2
dest=/etc/nginx/conf.d/www.webexample.com.conf
    notify: restart nginx

- name: Apply nginx configuration template
    template: src=nginx.conf.j2 dest=/etc/nginx/nginx.conf
    notify: restart nginx

- name: Start/Restart the Nginx service
    service: name=nginx state=started enabled=yes
[root@ip-192-168-30-11 tasks]#

[root@ip-192-168-30-11 templates]# pwd
/etc/ansible/roles/webservers/templates
[root@ip-192-168-30-11 templates]# cat nginx.conf.j2

user    nginx;
worker_processes    {{ ansible_processor_count }};

error_log    /var/log/nginx/error.log warn;
pid              /var/run/nginx.pid;



events {
       worker_connections    1024;
}
```

```
http {
    include       /etc/nginx/mime.types;
    default_type  application/octet-stream;

    log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '
                      '$status $body_bytes_sent "$http_referer" '
                      '"$http_user_agent" "$http_x_forwarded_for"';

    access_log  /var/log/nginx/access.log  main;

    sendfile        on;
    #tcp_nopush     on;

    keepalive_timeout  65;

    #gzip  on;

    include /etc/nginx/conf.d/*.conf;
}
[root@ip-192-168-30-11 templates]#

[root@ip-192-168-30-11 templates]# cat www.webexample.com.conf.j2
#
# Virtual host www.webexample.com
#
server
{
    server_name www.webexample.com;
    listen       {{ nginx_listening_port }} backlog={{ blparam }};
    root /var/www;
    index  index.html index.htm local.html;
```

```
        location /probe_local {
        alias /var/www;
        index local.html;
        allow all;
        }

        location /* {
                rewrite ^/* http://www.redhat.com/ redirect;
        }

        location /probe_remote {
                rewrite ^/probe_remote$ http://localhost:5500 redirect;
        }

        location /probe_listing {
                autoindex on;
                allow all;
                autoindex_exact_size off;
        }
}
[root@ip-192-168-30-11 templates]#
```