Load-Balancer et Initialisation (Keepalived et HAProxy) [OK]

Installation et Configuration de HAProxy :

Sur chaque master node

Installation

```
sudo apt -y install haproxy
sudo systemctl enable --now haproxy
```

Configuration :

`mv /etc/haproxy/haproxy.cfg /etc/haproxy/haproxy.cfg.bak && sudo nano /etc/haproxy/haproxy.cfg`

```
global
  maxconn 4096
  log /dev/log local0
  daemon
  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

listen stats
    bind *:9000
    stats enable
    stats uri /stats
    stats refresh 2s
    stats auth admin:statspasswordforhaproxy


# API Kubernetes : 6443 en TCP (TLS passthrough)
listen kubernetes-apiserver-https
    bind *:6443
    mode tcp
    option log-health-checks
    timeout client 10m
    timeout server 10m
    balance roundrobin
    server k8s-m1 192.168.20.240:6443 check
    server k8s-m2 192.168.20.241:6443 check
    server k8s-m3 192.168.20.242:6443 check
```

Redémarrage de HAProxy pour appliquer les changements `sudo systemctl restart haproxy`

Test si on écoute bien sur le port 6443 `sudo ss -lntp | grep 6443`

Installation et Configuration de Keepalived :

Sur chaque master node

Installation

`sudo apt -y install keepalived`

Configuration

Pensez a adapter la configuration en fonction de l'hote en baissant la priorité et en modifiant le nom de l'interface si nécéssaire.

`sudo nano /etc/keepalived/keepalived.conf`

```
vrrp_script chk_haproxy {
  script "pidof haproxy"
  interval 2
  weight -50
}

vrrp_instance VI_API {
  state BACKUP
  interface ens18              # <-- remplace par ton NIC
  virtual_router_id 51
  priority 150                 # M1:150, M2:120, M3:100
  advert_int 1
  authentication {
    auth_type PASS
    auth_pass changeMe
  }
  virtual_ipaddress {
    192.168.20.100/24 dev ens18 # <-- ton VIP
  }
```

```
  track_script { chk_haproxy }

  # Bascules ARP propres (options documentées dans le man keepalived)
  garp_master_delay 1
  garp_master_repeat 5
  garp_lower_prio_no_advert
}
```

Activation et vérification de l'installation :

```
sudo systemctl enable --now keepalived
journalctl -u keepalived -n 50 --no-pager
ip a | grep -A2 192.168.20.100
```

Ont peut également vérifié que le port fonctionne bien :

`nc -vz 192.168.20.100 6443`

Initialisation du Cluster Kubernetes :

```
sudo kubeadm init \
  --control-plane-endpoint 192.168.20.100:6443 \
  --pod-network-cidr 10.244.0.0/16 \
  --upload-certs
```

Une fois le conteneur initialisé, mettez vous avec votre utilisateur classique et executer ça (comme recommandé suite a l'installation) :

```
  mkdir -p $HOME/.kube
  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

Et si vous êtes en "root" alors cette commande ci : `export KUBECONFIG=/etc/kubernetes/admin.conf`

Revision #17
Created 9 October 2025 07:06:19 by Antonin POMIES
Updated 12 October 2025 17:43:49 by Antonin POMIES