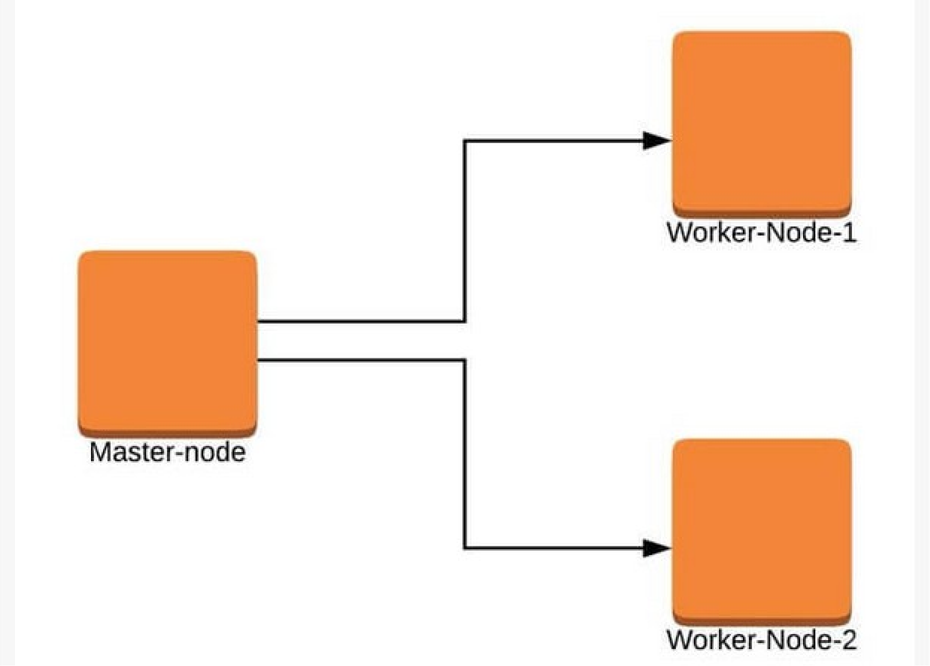
Redis Sentinel Based HA cluster in Kubernetes

For setting up the system we starting to install the necesary requirements we installed three VM with Centos 8 one master and 2 two slaves then we proceed to install docker and kubernetes in all three machines configure the the basic of the operating systems updated the system so at the end we have that works similar to this image:



We followed these tutorials to achieve the desired results:

https://docs.docker.com/engine/install/centos/

https://www.tecmint.com/setup-redis-high-availability-with-sentinel-in-centos-8/

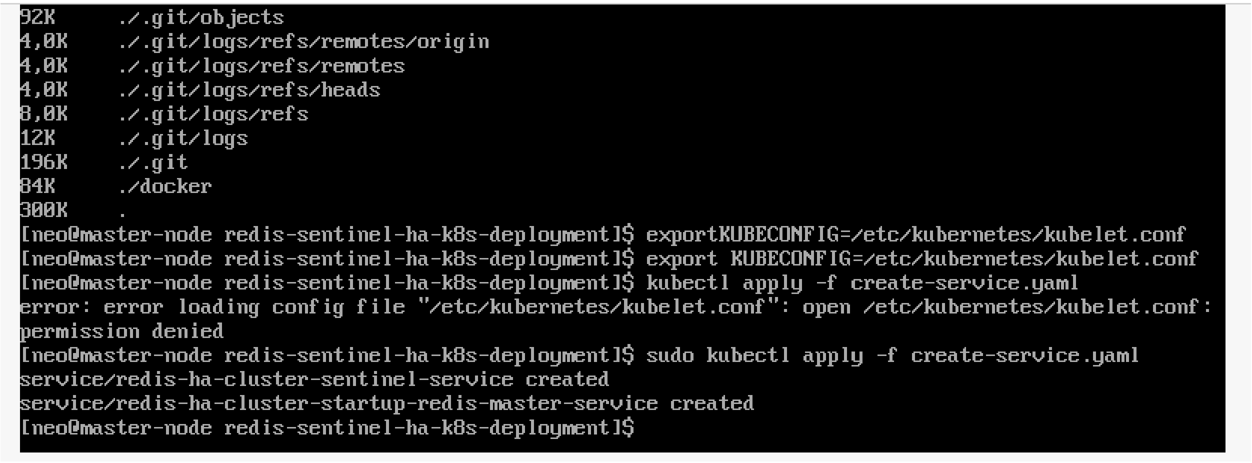
https://upcloud.com/community/tutorials/install-kubernetes-cluster-centos-8/

After have the system prepared we proceed to deploy our Redis cluster to kubernetes:

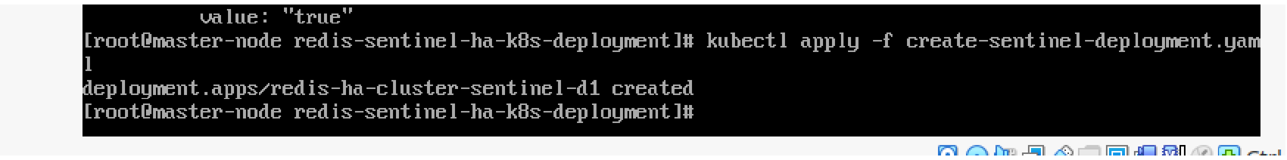
redis-sentinel-ha-k8s-deployment

We use git clone to colone the following repositiroy that has the deployment of all nodes already automated: https://github.com/sarweshsuman/redis-sentinel-ha-k8s-deployment.git

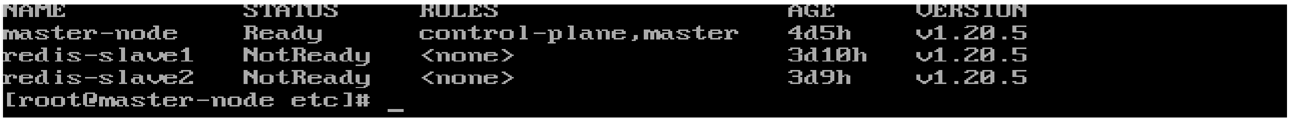
Then following the tutorial **https://sarweshsuman-1.medium.com/deploying-redis-ha-cluster-in-kubernetes-437162337625** we proceed to deploy the redis to kubernetes and here are some images of our own cluster:



```
92K     ./.git/objects
4,0K    ./.git/logs/refs/remotes/origin
4,0K    ./.git/logs/refs/remotes
4,0K    ./.git/logs/refs/heads
8,0K    ./.git/logs/refs
12K     ./.git/logs
196K    ./.git
84K     ./docker
300K    .
[neo@master-node redis-sentinel-ha-k8s-deployment]$ exportKUBECONFIG=/etc/kubernetes/kubelet.conf
[neo@master-node redis-sentinel-ha-k8s-deployment]$ export KUBECONFIG=/etc/kubernetes/kubelet.conf
[neo@master-node redis-sentinel-ha-k8s-deployment]$ kubectl apply -f create-service.yaml
error: error loading config file "/etc/kubernetes/kubelet.conf": open /etc/kubernetes/kubelet.conf:
permission denied
[neo@master-node redis-sentinel-ha-k8s-deployment]$ sudo kubectl apply -f create-service.yaml
service/redis-ha-cluster-sentinel-service created
service/redis-ha-cluster-startup-redis-master-service created
[neo@master-node redis-sentinel-ha-k8s-deployment]$
```



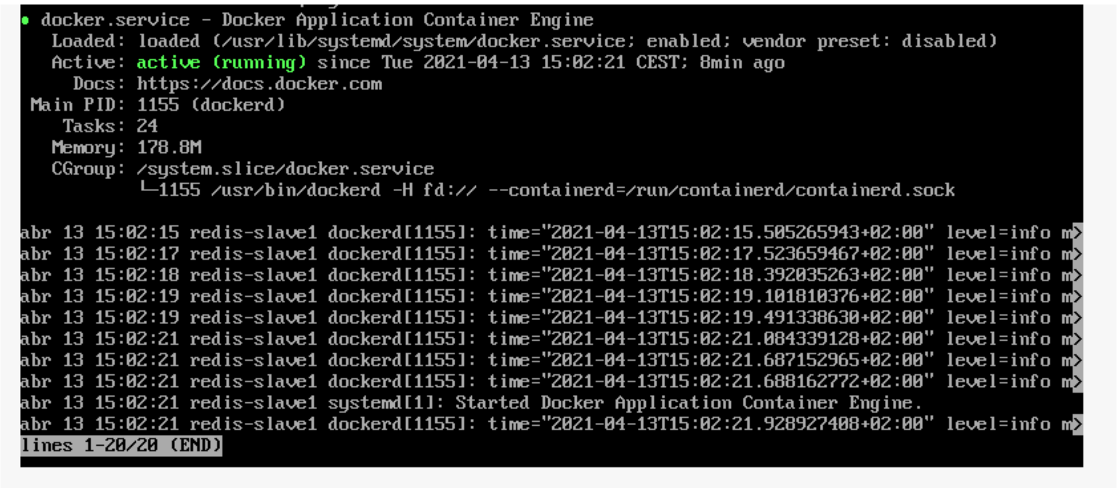
```
          value: "true"
[root@master-node redis-sentinel-ha-k8s-deployment]# kubectl apply -f create-sentinel-deployment.yam
l
deployment.apps/redis-ha-cluster-sentinel-d1 created
[root@master-node redis-sentinel-ha-k8s-deployment]#
```



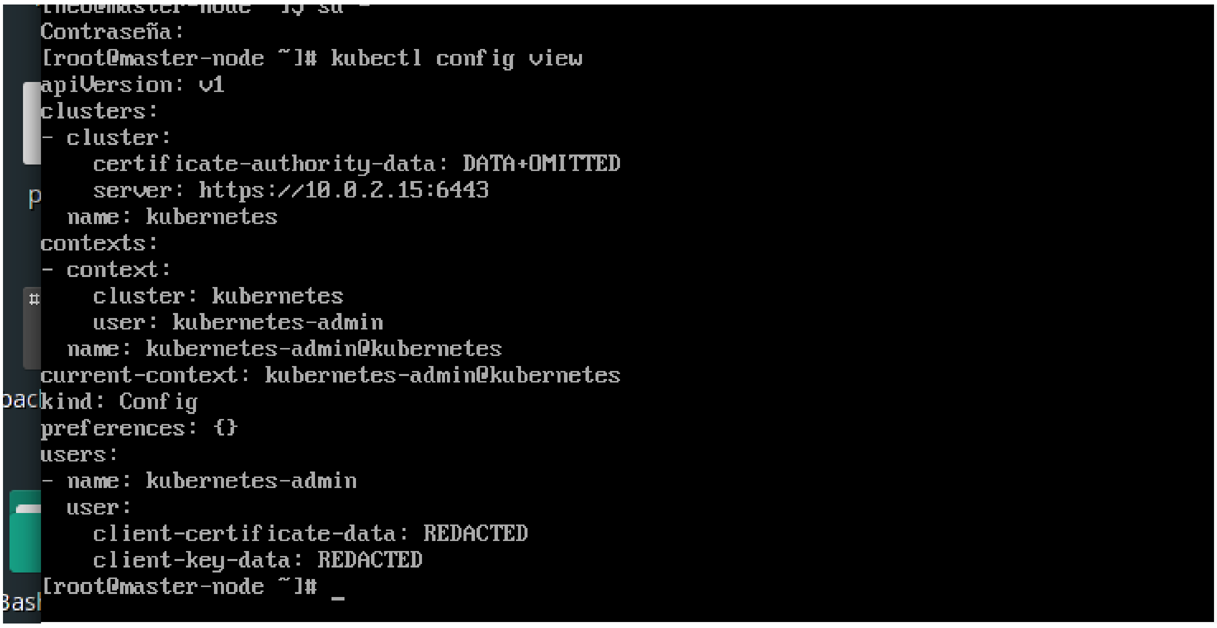
```
NAME           STATUS     ROLES                  AGE     VERSION
master-node    Ready      control-plane,master   4d5h    v1.20.5
redis-slave1   NotReady   <none>                 3d10h   v1.20.5
redis-slave2   NotReady   <none>                 3d9h    v1.20.5
[root@master-node etc]# _
```

During the installation and configuration of the system we encounter various issues or errors that we solve along the way, the most important was the lack of internet connection when the containers were started researching the matter we discovery that lacks of disk space cause this behaviour, increase solved the problem.

Aqui podemos apreciar el estado del sistema de contenedores:



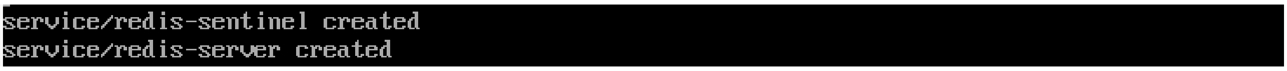
```
● docker.service - Docker Application Container Engine
   Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; vendor preset: disabled)
   Active: active (running) since Tue 2021-04-13 15:02:21 CEST; 8min ago
     Docs: https://docs.docker.com
 Main PID: 1155 (dockerd)
    Tasks: 24
   Memory: 178.8M
   CGroup: /system.slice/docker.service
           └─1155 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/containerd.sock

abr 13 15:02:15 redis-slave1 dockerd[1155]: time="2021-04-13T15:02:15.505265943+02:00" level=info m>
abr 13 15:02:17 redis-slave1 dockerd[1155]: time="2021-04-13T15:02:17.523659467+02:00" level=info m>
abr 13 15:02:18 redis-slave1 dockerd[1155]: time="2021-04-13T15:02:18.392035263+02:00" level=info m>
abr 13 15:02:19 redis-slave1 dockerd[1155]: time="2021-04-13T15:02:19.101810376+02:00" level=info m>
abr 13 15:02:19 redis-slave1 dockerd[1155]: time="2021-04-13T15:02:19.491338630+02:00" level=info m>
abr 13 15:02:21 redis-slave1 dockerd[1155]: time="2021-04-13T15:02:21.084339128+02:00" level=info m>
abr 13 15:02:21 redis-slave1 dockerd[1155]: time="2021-04-13T15:02:21.687152965+02:00" level=info m>
abr 13 15:02:21 redis-slave1 dockerd[1155]: time="2021-04-13T15:02:21.688162772+02:00" level=info m>
abr 13 15:02:21 redis-slave1 systemd[1]: Started Docker Application Container Engine.
abr 13 15:02:21 redis-slave1 dockerd[1155]: time="2021-04-13T15:02:21.928927408+02:00" level=info m>
lines 1-20/20 (END)
```



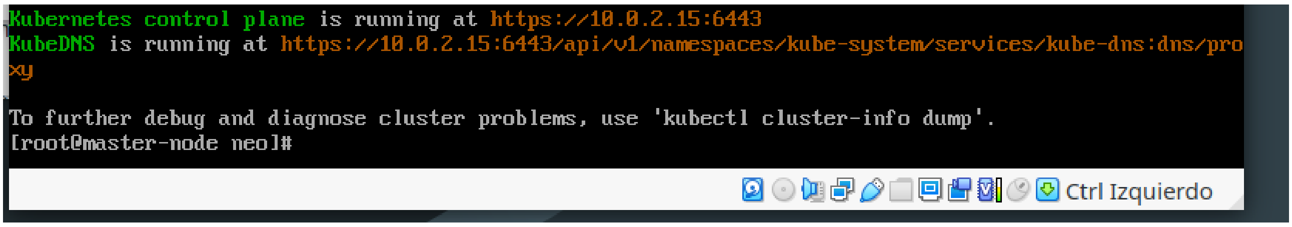
```
[neo@master-node ~]$ su -
Contraseña:
[root@master-node ~]# kubectl config view
apiVersion: v1
clusters:
- cluster:
    certificate-authority-data: DATA+OMITTED
    server: https://10.0.2.15:6443
  name: kubernetes
contexts:
- context:
    cluster: kubernetes
    user: kubernetes-admin
  name: kubernetes-admin@kubernetes
current-context: kubernetes-admin@kubernetes
kind: Config
preferences: {}
users:
- name: kubernetes-admin
  user:
    client-certificate-data: REDACTED
    client-key-data: REDACTED
[root@master-node ~]# _
```

Deployment of Redis

After install Kubernetes we proceed to use another github repositories to deploy our redis sentinel as seen in the image below:

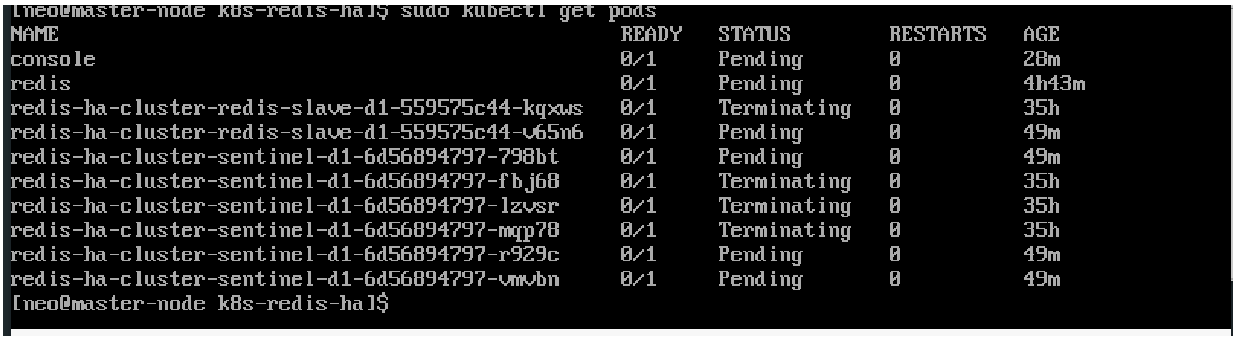


```
service/redis-sentinel created
service/redis-server created
```



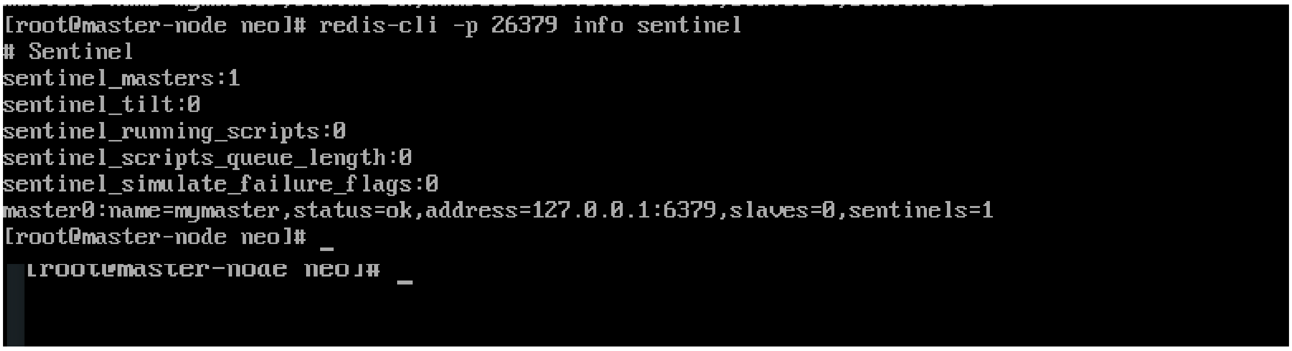
```
Kubernetes control plane is running at https://10.0.2.15:6443
KubeDNS is running at https://10.0.2.15:6443/api/v1/namespaces/kube-system/services/kube-dns:dns/pro
xy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
[root@master-node neo]#
```

Because of lack of resources in my machine could not continue to implement the systems kubernetes ecosystems as the pods staying in peding status instead of running as the image below represents:

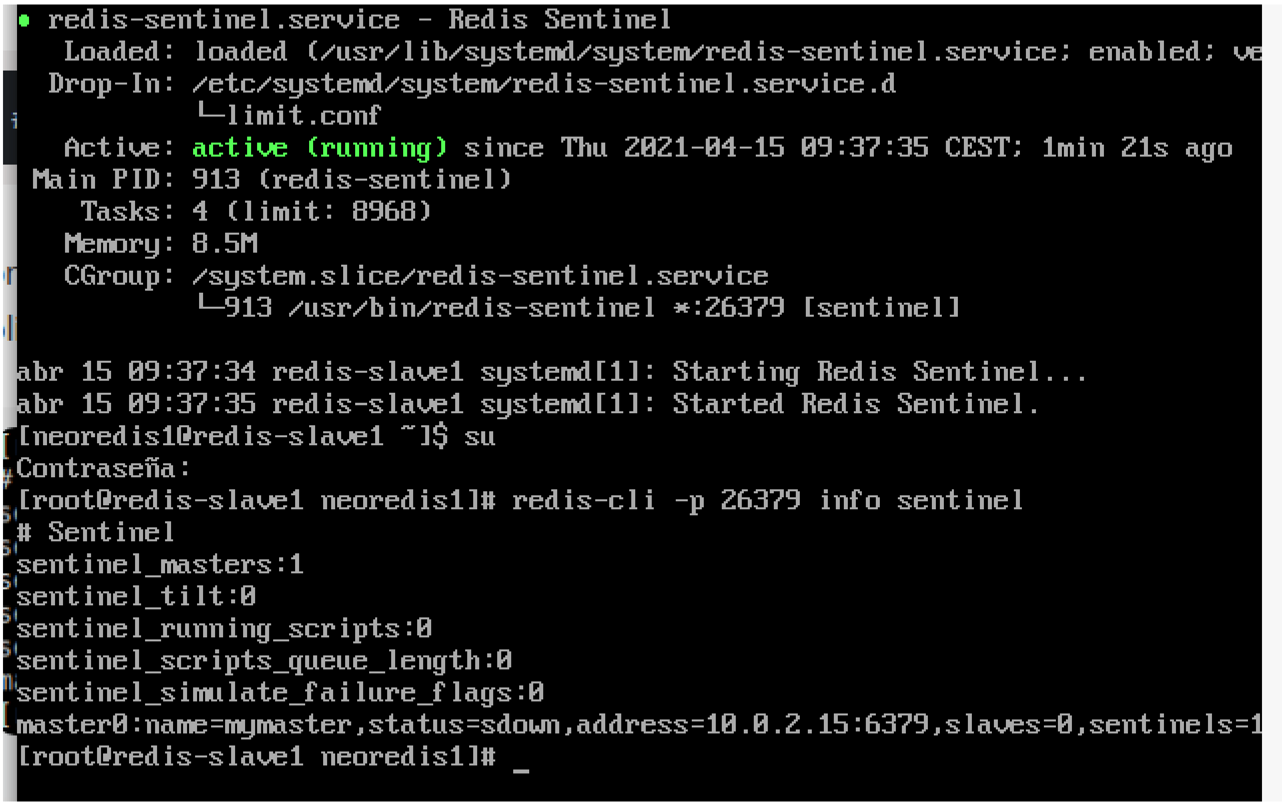


```
[neo@master-node k8s-redis-ha]$ sudo kubectl get pods
NAME                                           READY   STATUS        RESTARTS   AGE
console                                        0/1     Pending       0          28m
redis                                          0/1     Pending       0          4h43m
redis-ha-cluster-redis-slave-d1-559575c44-kqxws   0/1     Terminating   0          35h
redis-ha-cluster-redis-slave-d1-559575c44-v65n6   0/1     Pending       0          49m
redis-ha-cluster-sentinel-d1-6d56894797-798bt     0/1     Pending       0          49m
redis-ha-cluster-sentinel-d1-6d56894797-fbj68     0/1     Terminating   0          35h
redis-ha-cluster-sentinel-d1-6d56894797-lzvsr     0/1     Terminating   0          35h
redis-ha-cluster-sentinel-d1-6d56894797-mqp78     0/1     Terminating   0          35h
redis-ha-cluster-sentinel-d1-6d56894797-r929c     0/1     Pending       0          49m
redis-ha-cluster-sentinel-d1-6d56894797-vmvbn     0/1     Pending       0          49m
[neo@master-node k8s-redis-ha]$
```

As an alternative after hours of troubleshooting i moved to implement redis-sentinel on native systems a.k.a a centos server without containers



```
[root@master-node neo]# redis-cli -p 26379 info sentinel
# Sentinel
sentinel_masters:1
sentinel_tilt:0
sentinel_running_scripts:0
sentinel_scripts_queue_length:0
sentinel_simulate_failure_flags:0
master0:name=mymaster,status=ok,address=127.0.0.1:6379,slaves=0,sentinels=1
[root@master-node neo]# _
 [root@master-node neo]# _
```



```
● redis-sentinel.service - Redis Sentinel
   Loaded: loaded (/usr/lib/systemd/system/redis-sentinel.service; enabled; ve
  Drop-In: /etc/systemd/system/redis-sentinel.service.d
           └─limit.conf
   Active: active (running) since Thu 2021-04-15 09:37:35 CEST; 1min 21s ago
 Main PID: 913 (redis-sentinel)
    Tasks: 4 (limit: 8968)
   Memory: 8.5M
   CGroup: /system.slice/redis-sentinel.service
           └─913 /usr/bin/redis-sentinel *:26379 [sentinel]

abr 15 09:37:34 redis-slave1 systemd[1]: Starting Redis Sentinel...
abr 15 09:37:35 redis-slave1 systemd[1]: Started Redis Sentinel.
[neoredis1@redis-slave1 ~]$ su
Contraseña:
[root@redis-slave1 neoredis1]# redis-cli -p 26379 info sentinel
# Sentinel
sentinel_masters:1
sentinel_tilt:0
sentinel_running_scripts:0
sentinel_scripts_queue_length:0
sentinel_simulate_failure_flags:0
master0:name=mymaster,status=sdown,address=10.0.2.15:6379,slaves=0,sentinels=1
[root@redis-slave1 neoredis1]# _
```

Documentation not finished….

Virtualbox access credentials:

redis-master1 same for root user:

user: neo

pass: lunes123

redis-slave1 same for root user:

user:neoredis1

pass: lunes123

redis-slave2 same for root user:

user: neo-redis1

pass:lunes123