

# HA / CLUSTER

- [Keepalived HA](#)
- [Verifier l'état d'un cluster sous PaceMaker](#)
- [Correction d'erreur drbd resynchronisation de la réplication](#)
- [HAproxy.conf](#)
- [Commande Tshoot Cluster NFS/PACEMAKER/COROSYNC](#)
- [Script de Nettoyage ressource sur un cluster NFS](#)
- [Patroni cluster](#)

# Keepalived HA

Nous avons setup un Keepalived pour les base de donnée avec une VIP (virtual adresse IP), pour qu'en cas de shutdown sur la base de donnée MAITRE, le standby reprennent directement le relais

Configuration numéro 1, MAITRE dans /etc/keepalived/keepalived.conf



```
root@Test_HA_pgSQL-02:/var/lib/pgsql/14/data
vrrp_script check_postgresql {
    script "/usr/local/bin/check_postgresql.sh"
    interval 1
    timeout 2
    rise 2
    fall 2
}

vrrp_instance VI_1 {
    state MASTER
    interface ens192
    virtual_router_id 51
    priority 255
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 19901
    }

    virtual_ipaddress {
        192.168.1.234
    }
    track_script {
        check_postgresql
    }
}
```

Configuration numéro 2, SLAVE dans /etc/keepalived/keepalived.conf



Une meilleur config pour apache 1 / master - check sur le service httpd.

```
#MASTER#

global_defs {
}

vrrp_script chk_httpd {
    script "/usr/bin/killall -0 httpd" # widely used idiom
    interval 15 # check every 2 seconds
    weight 2 # add 2 points of prio if OK
}

vrrp_instance VI_1 {
    interface ens192
    state MASTER # or "BACKUP" on backup
    priority 101 # 101 on master, 100 on backup
    virtual_router_id 51
    authentication {
        auth_type PASS
        auth_pass [REDACTED]
    }
    virtual_ipaddress {
        [REDACTED].17.106/27
    }
    unicast_src_ip [REDACTED].17.99/27 # This haproxy node
    unicast_peer {
        [REDACTED].17.103/27 # Other haproxy nodes
    }
    track_script {
        chk_httpd
    }
}
```

Une meilleur config pour apache 2 / backup - check sur le service httpd.

```
global_defs {
}

vrrp_script chk_httpd {
    script "/usr/bin/killall -0 httpd" # widely used idiom
    interval 15 # check every 2 seconds
    weight 2 # add 2 points of prio if OK
}

vrrp_instance VI_1 {
    interface ens192
    state BACKUP # or "BACKUP" on backup
    priority 100 # 101 on master, 100 on backup
    virtual_router_id 51
    authentication {
        auth_type PASS
        auth_pass [REDACTED]
    }
    virtual_ipaddress {
        [REDACTED].17.106/27
    }
    unicast_src_ip [REDACTED].17.103/27 # This haproxy node
    unicast_peer {
        [REDACTED].17.99/27 # Other haproxy nodes
    }
    track_script {
        chk_httpd
    }
}
```

# Verifier l'état d'un cluster sous PaceMaker

```
[root@ :~]
# pcs status corosync
Error: corosync not running
[root@ :~]
# pcs cluster start
Starting Cluster (corosync)...
Starting Cluster (pacemaker)...
[root@ :~]
# pcs status
Cluster name: 
Stack: corosync
Current DC: NONE
Last updated: Tue Oct  4 09:03:21 2022
Last change: Sun Sep  4 19:34:56 2022 by hacluster via crm_attribute on frd825d17dbl

2 nodes configured
6 resource instances configured

OFFLINE: [ ]

Full list of resources:

Master/Slave Set: datamaster [dataraw]
  Stopped: [ ]
Resource Group: MYSQL
  FS_DataBDD (ocf::heartbeat:Filesystem):      Stopped
  cluster_admin_VIP (ocf::heartbeat:IPaddr2):    Stopped
  cluster_mysql_VIP (ocf::heartbeat:IPaddr2):    Stopped
  Service_MariaDB (ocf::heartbeat:mysql): Stopped

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
```

```
[root@ :~]
# pcs status corosync

Membership information
-----
Nodeid      Votes Name
    1         1   db1
    2         1   db1 (local)
```

Pour l'alerte suivante : App-DRBD-Monitor --> DRBD: 1 crit, 0 okay:

Le service drbd ne doit pas être démarré manuellement : il est géré par pacemaker via les commandes pcs

```
# service drbd status
• drbd.service - LSB: Control DRBD resources.
  Loaded: loaded (/etc/init.d/drbd; generated)
  Active: inactive (dead)
  Docs: man:systemd-sysv-generator(8)
```

Identifier l'état du cluster :

`drbdadm status`

`drbdmon` (ne doit pas afficher d'alerte )

Mettre les sonde drbd en maintenance et procéder à une resynchronisation drbd

```
[root@redhat:~]
# pcs status
Cluster name: redhat
Stack: corosync
Current DC: redhat-dbl (version 1.1.23-1.el7_9.1-9acf116022) - partition with quorum
Last updated: Wed Oct 11 16:07:18 2023
Last change: Wed Oct 11 14:24:19 2023 by root via cibadmin on frd825p29dbl

2 nodes configured
6 resource instances configured

Online: [ redhat-dbl ]

Full list of resources:

Master/Slave Set: datamaster [dataraw]
  Masters: [ redhat-dbl ]
  Slaves: [ redhat-dbl ]
Resource Group: MYSQL
  FS_DataBDD (ocf::heartbeat:Filesystem): Started redhat-dbl
  cluster_admin_VIP (ocf::heartbeat:IPaddr2): Stopped
  cluster_mysql_VIP (ocf::heartbeat:IPaddr2): Stopped
  Service_MariaDB (ocf::heartbeat:mysql): Stopped

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
```

`pcs resource enable MYSQL`

```
[root@redhat:~]
# pcs status
Cluster name: redhat MYSQL
Stack: corosync
Current DC: redhat-dbl (version 1.1.23-1.el7_9.1-9acf116022) - partition with quorum
Last updated: Wed Oct 11 16:09:20 2023
Last change: Wed Oct 11 14:24:19 2023 by root via cibadmin on redhat-dbl

2 nodes configured
6 resource instances configured

Online: [ redhat-dbl ]

Full list of resources:

Master/Slave Set: datamaster [dataraw]
  Masters: [ redhat-dbl ]
  Slaves: [ redhat-dbl ]
Resource Group: MYSQL
  FS_DataBDD (ocf::heartbeat:Filesystem): Started redhat-dbl
  cluster_admin_VIP (ocf::heartbeat:IPaddr2): Started redhat-dbl
  cluster_mysql_VIP (ocf::heartbeat:IPaddr2): Started redhat-dbl
  Service_MariaDB (ocf::heartbeat:mysql): Started frd825p30dbl

Daemon Status:
  corosync: active/enabled
  pacemaker: active/enabled
  pcsd: active/enabled
```

Déplacement des ressources action préreboot/maj :

Vérifier l'état du cluster

`pcs status --full drbdadm status`

`drbdmon` (ne doit pas afficher d'alerte )

Si drbd ok: déplacer les ressources sur le nœud sur lequel il n'y aura pas d'opération `pcs resource move NOM_DU_GROUP_DE_RESSOURCE` (exemple ci dessus : MYSQL )

Vérifier le déplacement des ressources avec `pcs status` mettre le nœud secondaire (sur lequel on va faire la mise à jour ou le reboot) en maintenance

`pcs node maintenance nomdunoeud`

mettre ensuite le nœud en standby

`pcs node standby nomdunoeud`

procéder aux opérations de maintenance

# Correction d'erreur drbd resynchronisation de la réplication

Correction d'erreur drbd resynchronisation de la réplication :  
Identifier le noeud maitre

```
pcs status --full
```

mettre le nœud secondaire en maintenance avec pcs

```
pcs node maintenance nomdunoeudpcs
```

Identifier la ou les ressources drbd avec la commande :

```
drbdadm dump
```

sur le nœud secondaire invalider les données de réplication drbd pour chacune des ressources :

```
drbdadm invalidate Nom_de_ressource_drbd
```

Sur le nœud primaire préparer pour une resynchronisation forcée

```
drbdadm invalidate-remote Nom_de_ressource_drbd
```

Sortir le nœud secondaire de maintenance

```
pcs node unmaintenance nomdunoeudpcs
```

Vérifier la synchronisation drbd sur les deux nœud avec d'un coté :

```
drbdmon
```

et sur l'autre nœud:

```
watch -n1 drbdadm status
```



# HAproxy.conf

Config sur les machines proxy

```
[root@ ~]# cat /etc/haproxy/haproxy.cfg
frontend pgsql_frontend
  bind *:5432
  mode tcp
  default_backend pgsql_backend

backend pgsql_backend
  mode tcp
  balance roundrobin
  option pgsql-check user postgres # Utilisation de l'utilisateur postgres pour la vérification
  server .16.25:5432 check port 8008
  server .16.26:5432 check port 8008
  server .16.8:5432 check port 8008
# option redispatch 1s
option httpchk
http-check send hdr Content-Type "application/json;charset=UTF-8" hdr Host . body "{\"id\": 3, \"role\": \"master\"}"

#frontend pgsql_write_frontend
# bind *:5433
# mode tcp
# default_backend pgsql_write_backend

#backend pgsql_write_backend
# mode tcp
# balance roundrobin
# option pgsql-check user postgres # Utilisation de l'utilisateur postgres pour la vérification
# server .16.25:5432 check port 8008
# server .16.26:5432 check port 8008
```

Format json pour vérification :

The screenshot shows a Firefox browser window with the address bar displaying `.16.8:8008/`. The page content is a JSON response, likely from a PostgreSQL server, showing the state of a replication slot. The JSON is displayed in the 'JSON' tab of the browser's developer tools. The data includes:

- `state`: "running"
- `postmaster_start_time`: "2024-03-21 14:18:11.476935+01:00"
- `role`: "replica"
- `server_version`: 140009
- `xlog`:
  - `received_location`: 36909875640
  - `replayed_location`: 36909875640
  - `replayed_timestamp`: null
  - `paused`: false
  - `timeline`: 48
  - `replication_state`: "streaming"
  - `dcs_last_seen`: 1711027244
  - `database_system_identifier`: "7268271095787782973"
- `patroni`:
  - `version`: "3.2.1"
  - `scope`: "CLUSTER"
  - `name`: "

The bottom of the JSON viewer shows a search bar with the text 'id' and 3 of 3 matches.

# Commande Tshoot Cluster NFS/PACEMAKER/COROSYNC

lors de l'installation de votre cluster il faut que cette requête soit en état OK sur les 2 nœuds sinon votre cluster ne basculeras pas, j'ai eu le problème sur mes proxy ouvert, j'ai du les désactivé pour que la bascule fonctionne:

```
curl -k https://****.fr:2224/remote/status/
```

Commande pour cleanup vos ressource si elles sont KO:

```
crm_resource --cleanup -r p_drbd_ha_nfs  
crm_resource --cleanup -r p_fs_drbd1
```

Commande pour disable ou remettre un noeud dans le cluster:

```
sudo pcs cluster standby <nom_du_nœud>  
sudo pcs cluster unstandby <nom_du_nœud>
```

disable les ressource sur un noeuds isolé:

```
pcs resource disable p_exportfs_openshift_prod  
pcs resource disable p_exportfs_prod  
pcs resource disable p_nfsserver  
pcs resource disable p_fs_drbd1  
pcs resource disable p_drbd_ha_nfs
```

```
crm_resource --resource p_drbd_ha_nfs --node ****.fr --demote
```

réactiver les ressource sur un noeuds isolé:

```
pcs resource enable p_drbd_ha_nfs  
pcs resource enable p_fs_drbd1  
pcs resource enable p_nfsserver  
pcs resource enable p_exportfs_prod  
pcs resource enable p_exportfs_openshift_prod
```

```
crm_resource --resource p_drbd_ha_nfs --node ****.fr --promote
```

deconnecter ou connecter drbd sur le noeud isolé

```
drbdadm disconnect ha_nfs
```

```
sudo drbdadm connect ha_nfs
```

Forcer la synchronisation avec les données du nœud primaire

```
drbdadm connect --discard-my-data ha_nfs
```

Status du drbd

```
drbdadm status
```

# Script de Nettoyage ressource sur un cluster NFS

Script à appliquer sur une crontab de 24h pour effectuer un nettoyage de vos ressource, si le NFS est consommé fréquemment, toujours choisir la première ressource de votre cluster cela nettoieras le reste à la suite peu importe le nombre de ressource derrière, le script rajoute un print sur des logs si cela a bien fonctionner :

```
#!/bin/bash

# Variable pour le fichier de log
LOG_FILE="/var/log/cleanup_p_drbd_ha_nfs.log"

# Bloc conditionnel pour exécuter le nettoyage
if sudo crm_resource -cleanup -r p_drbd_ha_nfs && sudo crm_resource -cleanup -r p_exportsfs_pprod; then
    # Si les deux commandes se sont exécutées avec succès
    echo "$(date +%Y-%m-%d %H:%M:%S) - Clean up successful" >> "$LOG_FILE"
else
    # Si au moins une des commandes a échoué
    echo "$(date +%Y-%m-%d %H:%M:%S) - Clean up failed" >> "$LOG_FILE"
fi
```

# Patroni cluster

Lister un cluster patroni :

patronictl -c /etc/patroni/patroni.yml list

```
[root@IDF-ge0-001 ~]# patronictl -c /etc/patroni/patroni.yml list
+ Cluster: IDFPPR-GE0-CLUSTER (7267900578268639473) -----+
| Member          | Host                | Role   | State   | TL | Lag in MB |
+-----+-----+
| IDF-ge0-001     | 10.242.13.15:5438   | Leader | running | 34 |           |
| IDF-ge0-002     | 10.242.13.16:5438   | Replica | streaming | 34 | 0         |
| IDF-ge0-003     | 10.242.13.55:5438   | Replica | streaming | 34 | 0         |
+-----+-----+
```