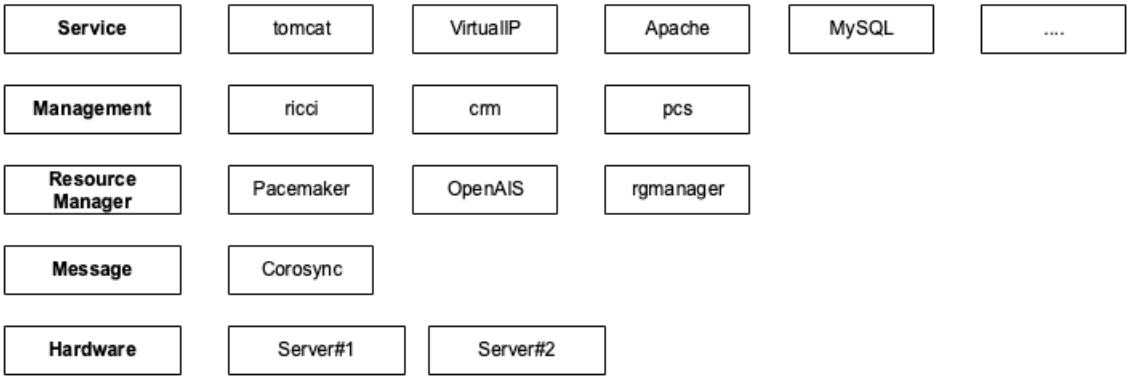
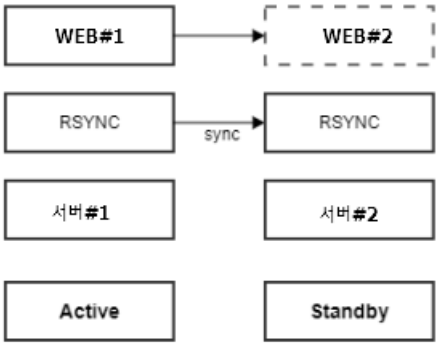
corosync / pacemaker를 이용한 HA구성

1. HA stack



    1. Hardware : 물리서버
    2. Message : Clustering 맴버간 통신상태 확인
    3. Resoure Manager : failover / failback 작업 수행
    4. Management : HA를 구성하도록 제공되는 Tool
    5. Service : HA용 서비스
2. Service Architecture (Active / Standby)



설치 방안 (모든 노드)

1.패키지 설치

```
$> yum install corosync pacemaker pcs automake -y
```

1.HA를 위한 사용자 계정설정 후 pcsd 서비스 실행 (모든노드 동일한 패스워드 입력)

```
$> passwd hacluster
Changing password for user hacluster.
New password:
BAD PASSWORD: it is based on a dictionary word
Retype new password:
passwd: all authentication tokens updated successfully.
$> systemctl start pcsd
```

1. 클러스터 인증구성 (이후 과정은 master or slave 중 1대에서만 수행)

```
$> pcs cluster auth master slave
Username: hacluster
Password:
master: Authorized
```

```
slave: Authorized
```

2. 클러스터 구성

```
$> pcs cluster setup --name cluster master slave
Destroying cluster on nodes: master, slave...
master: Stopping Cluster (pacemaker)...
slave: Stopping Cluster (pacemaker)...
slave: Successfully destroyed cluster
master: Successfully destroyed cluster

Sending 'pacemaker_remote authkey' to 'master', 'slave'
slave: successful distribution of the file 'pacemaker_remote authkey'
master: successful distribution of the file 'pacemaker_remote authkey'
Sending cluster config files to the nodes...
master: Succeeded
slave: Succeeded

Synchronizing pcsd certificates on nodes master, slave...
master: Success
slave: Success
Restarting pcsd on the nodes in order to reload the certificates...
master: Success
slave: Success
```

3. 클러스터 에 필요한 서비스 구동

```
$> pcs cluster start --all
# pcs status corosync

Membership information
----------------------
    Nodeid      Votes Name
         1          1 master (local)
         2          1 slave


#pcs property set stonith-enabled=false
```

4. 재부팅 후 서비스 활성화 (양쪽서버에서 모두 수행)

```
# systemctl enable pacemaker
# systemctl enable corosync
```

관리툴(crmsh) 설치

1. crmsh설치를 위한 OS 패키지 설치

```
$> yum install -y python-lxml  pacemaker-libs-devel asciidoc python-dateutil python-yaml --skip-broken
$> yum install python-devel python-setuptools -y
```

2. crmsh 패키지 다운로드 후 설치

```
$> ./autogen.sh
autoconf:       autoconf (GNU Autoconf) 2.63
automake:       automake (GNU automake) 1.11.1
aclocal
automake --add-missing --include-deps --copy
configure.ac:33: installing `./install-sh'
configure.ac:33: installing `./missing'
autoconf
Now run ./configure

>#  ./configure
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for a thread-safe mkdir -p... /bin/mkdir -p
checking for gawk... gawk
checking whether make sets $(MAKE)... yes
```

```
checking for python... /usr/bin/python
checking for python version... 2.6
checking for python platform... linux2
checking for python script directory... ${prefix}/lib/python2.6/site-packages
checking for python extension module directory... ${exec_prefix}/lib64/python2.6/site-packages
checking for asciidoc... no
configure: creating ./config.status
config.status: creating Makefile
config.status: creating hb_report/hb_report
config.status: creating crm.conf
config.status: creating version
>#  make; make install
...
```

3. CRM 라이브러리 연동

```
# ln -s /usr/local/lib/python2.6/site-packages/crmsh   /usr/lib/python2.6/site-packages/
```

4. 서비스 구동

```
# /etc/init.d/corosync start
>#  /etc/init.d/pacemaker start
>#  chkconfig corosync on
>#  chkconfig pacemaker on
```

- 주의사항 : Pacemaker/Corosync를 이용한 HA시 corosync에서 메시지를 처리하기 때문에 pacemaker보다 corosync가 먼저 실행되어야 한다.
- 서비스 중지는 pacemaker → corosync 순으로 종료

5. 서비스 이상여부 확인

```
$> crm_mon -1
```

6. HA 리소스 구성

```
########## 동일한 가중치 설정으로 Auto-Failback 비활성화 #############
>#crm configure
crm(configure)# property stonith-enabled=false
crm(configure)# property no-quorum-policy=ignore
crm(configure)# rsc_defaults resource-stickiness=100
crm(configure)# commit



############## HA용 VIP 설정 ####################
crm(configure)# primitive TEST_STOR_VIP ocf:heartbeat:IPaddr2 params ip={HA VIP} cidr_netmask=24 op monitor interval=10s

############## DRBD 연동 ##########################
crm(configure)# primitive TEST_STOR_DRBD ocf:linbit:drbd params drbd_resource="drbd0" op monitor interval="20s" role="Master" op monitor interval="20s" role="Slave"
crm(configure)# ms MS_TEST_STOR_DRBD TEST_STOR_DRBD meta master-max="1" master-node-max="1" clone-max="2" clone-node-max="1" notify="true"
crm(configure)# primitive TEST_STOR_FS ocf:heartbeat:Filesystem params device="/dev/drbd0" directory="/data" fstype="xfs"




############ DRBD 그룹구성, 서비스 구동 순서 설정 #################3
crm(configure)# group TEST_STOR_GP TEST_STOR_FS TEST_STOR_VIP meta migration-threshold="5"
crm(configure)# colocation DRBD_on_HA inf: TEST_STOR_GP MS_TEST_STOR_DRBD:Master
crm(configure)# order DRBD_after_HA inf: MS_TEST_STOR_DRBD:promote TEST_STOR_GP:start




############## 리소스 구성후 완료 ######################
crm(configure)# commit
```

7. HA구성정보 확인

```
# crm_mon -1
Stack: classic openais (with plugin)
```

```
Current DC: TEST-STOR#1 (version 1.1.18-3.el6-bfe4e80420) - partition with quorum
Last updated: Wed Dec 18 18:29:46 2019
Last change: Mon Sep  2 19:26:34 2019 by root via crm_resource on TEST-STOR#1

2 nodes configured (2 expected votes)
5 resources configured

Online: [ TEST-STOR#1 TEST-STOR#2 ]

Active resources:

 Master/Slave Set: MS_TEST_STOR_DRBD [TEST_STOR_DRBD]
     Masters: [ TEST-STOR#1 ]
     Slaves: [ TEST-STOR#2 ]
 Resource Group: TEST_STOR_GP
     TEST_STOR_FS        (ocf::heartbeat:Filesystem):    Started TEST-STOR#1
     TEST_STOR_APACHE    (lsb:httpd):    Started TEST-STOR#1
     TEST_STOR_VIP       (ocf::heartbeat:IPaddr):        Started TEST-STOR#1
```

Revision #4
★Created 7 June 2022 01:20:51 by artop0420
Updated 24 December 2023 02:31:24 by artop0420