목차

Fence Death 상황을 방지하기 위한 2노드 클러스터 설정 3
RHEL6, RHEL7, RHEL8 pacemaker 3
RHEL5 ,RHEL6 cman 3
pacemaker 2 node 클러스터 설정 4
참조링크 5

Fence Death 상황을 방지하기 위한 2노드 클러스터 설정

Fence Death 상황이랑 2개의 노드가 Split Brain 이든 뭐든 특정 장애 상황에 서로를 Fence 작동하는 상황을 말한다.

한마디로 서로의 장비를 리부팅 시키는 것이다.

이런경우 어느 한쪽의 fence 동작을 몇초정도 딜레이 시킴으로서 해결이 가능하다.

RHEL6, RHEL7, RHEL8 pacemaker

한쪽 노드의 STONITH 장치의 delay값을 설정한다.

```
# pcs stonith create <name> <agent> [options] delay=<seconds>
# Example:
$ pcs stonith create node1-fence fence_ipmilan ipaddr=node1-ilo.example.com
login=fenceuser passwd=S3cur3p455&! pcmk_host_list="node1.example.com"
delay=5
$ pcs stonith create node2-fence fence_ipmilan ipaddr=node2-ilo.example.com
login=fenceuser passwd=S3cur3p455&! pcmk_host_list="node2.example.com"
```

RHEL5 ,RHEL6 cman

한쪽노드의 fencedevice 설정에 `delay=5` 설정을 넣어주면 해결된다.

1. 아래는 클러스터 노드의 펜싱 설정에서 delay설정을 하는방법

```
<clusternode name="node1.example.com" nodeid="1" votes="1">
    <fence>
        <method name="1">
            <device name="Vmware" port="node1" uuid="4223dbb3-6ec6-fg65-
ef4c-dhu7562dff56cf" delay="10"/>
        </method>
    </fence>
</clusternode>
```

2. 아래는 fencedevice 설정에서 delay를 설정하는 방법

```
   <fencedevices>
       <!--
           <fencedevice name="<name>" agent="<agent>" ipaddr="<ip or
```

```
hostname>" [... parameters ...] delay="<seconds>"/>
            Example:
       -->
       <fencedevice name="node1-virt" agent="fence_xvm" port="vm-node1"
delay="5" />
   </fencedevices>
```

둘중 어느 한 방법을 사용하기만 하면 된다.

pacemaker 2 node 클러스터 설정

클러스터에 노드가 2개뿐인경우 `/etc/corosync/corosync.conf`의 `quorum` 섹션에서 `two_node`및 `wait_for_all`옵션이 활성화되어 있는지 확인합니다

> 만약 `two_node`가 활성화 되어있고 `wait_for_all`이 설정되어있지 않은경우 `wait_for_all`은 자동으로 활성화 됩니다. `wait_for_all`을 지정하는경우 지정된 값이 적용됩니다.

```
# # wait_for_all is enabled implicitly because two_node is enabled
# # two_node가 활성화 되어있으므로 wait_for_all은 자동으로 활성화 됩니다.
quorum {
    provider: corosync_votequorum
    two_node: 1
}

# # wait_for_all is enabled explicitly
# # wait_for_all 을 명시적으로 활성화 합니다.
quorum {
    provider: corosync_votequorum
    two_node: 1
    wait_for_all: 1
}

# # wait_for_all is disabled because two_node is not enabled
# # two_node가 활성화되어있지 않으므로 wait_for_all 도 자동으로 비활성화 됩니다.
# # wait_for_all is not set explicitly so it defaults to 0 here
# # wait_for_all이 명시적으로 선언되어있지 않으므로 기본값은 비활성입니다.
quorum {
    provider: corosync_votequorum
}

# # wait_for_all is disabled explicitly
# # wait_for_all 을 명시적으로 비활성화 합니다.
quorum {
```

```
    provider: corosync_votequorum
    two_node: 1
    wait_for_all: 0
}
```

위의 내용을 적용하고 클러스터 설정을 동기화 합니다.

```
$ pcs cluster sync
```

모든 노드의 클러스터 서비스를 재기동 합니다. 아무 노드에서나 한군데서 실행하면 양쪽 다 반영됩니다.

```
$ pcs cluster stop --all
$ pcs cluster start --all
```

참조링크

- https://access.redhat.com/solutions/54829
- https://access.redhat.com/solutions/3565071
- https://access.redhat.com/solutions/1294873

From:
https://atl.kr/dokuwiki/ - **AllThatLinux!**

Permanent link:
https://atl.kr/dokuwiki/doku.php/fence_death_%EC%83%81%ED%99%A9%EC%9D%84_%EB%B0%A9%EC%A7%80%ED%95%98%EA%B8%B0_%EC%9C%84%ED%95%9C_2%EB%85%B8%EB%93%9C_%ED%81%B4%EB%9F%AC%EC%8A%A4%ED%84%B0_%EC%84%A4%EC%A0%95

Last update: **2024/01/18 07:56**