Introducción (I)

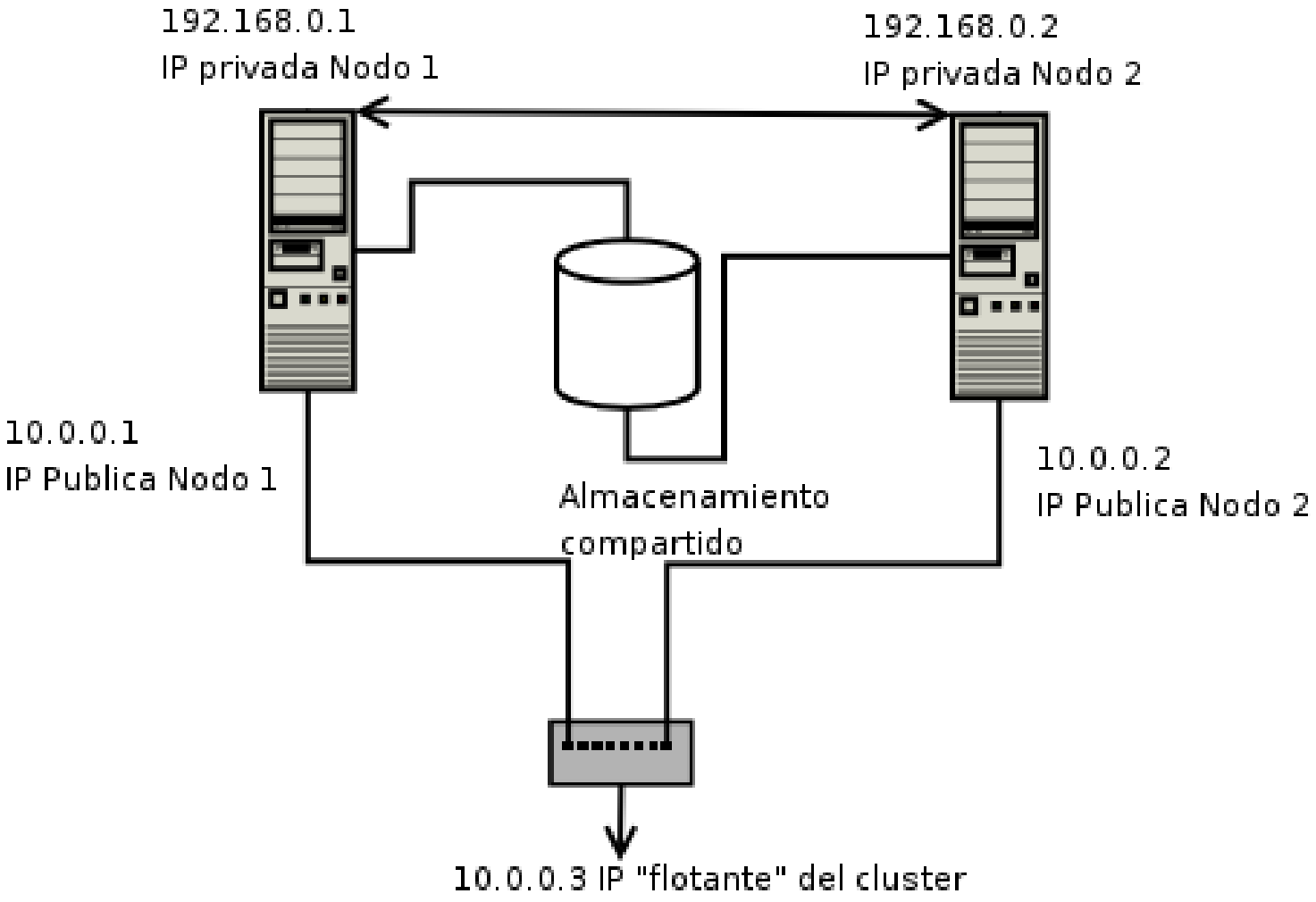
- Un sistema de alta disponibilidad es aquel que es capaz de seguir ofreciendo sus servicios ante determinados fallos hardware y/o software.
- Se basan en:
  - Un conjunto de nodos que proporcionan los servicios del cluster.
  - Un conjunto de estrategias para migrar los servicios entre los nodos.
  - Un servicio de comunicación entre los nodos.
  - Un sistema para detectar errores en el servidor.
- Generalmente incluyen un sistema de almacenamiento compartido: DRBD.

Introducción (II)

- Unos paquetes de software que permite montar sistemas en alta disponibilidad es pcs/corosync/pacemaker.
  - Pcs proporciona la autenticación y control de los nodos.
  - Corosync proporciona la comunicación entre los nodos.
  - Pacemaker proporciona el control de los recursos en cada uno de los nodos del cluster.
- Los servicios se arrancan con los comandos:

```
Systemctl start pcsd.service
systemctl start corosync.service
systemctl start pacemaker.service
```

Introducción (III)



El comando pcs

- Permite configurar y manejar el cluster.

Opción	Descripción
resource	Maneja los recursos del cluster
cluster	Maneja y configura las opciones del cluster y de los nodos del cluster.
stonith	Configura la politica de desactivación y reinicio de un nodo cuando el cluster considera que funciona mal.
property	Asigna las propiedades de pacemaker.
constraint	Asigna las restricciones existentes entre los recursos del cluster.
status	Muestra el estado del cluster.
config	Muestra la configuración completa del cluster.

Pcsd (I)

- Se debe asignar una contraseña a un usuario:
  - Será el que ejecute el servicio de pacemaker.
  - Debe ser el mismo en todos los nodos del cluster y con la misma contraseña.
  - De forma general se utiliza el usuario hacluster.

```
passwd hacluster
```

  - Si se desea utilizar otro usuario es necesario modificar varios ficheros de configuración.

Pcsd (II)

- Los nodos del cluster se crean con el comando:

```
pcs cluster auth nodo1 nodo2
```

  - Solicitando el usuario del cluster y su contraseña.

```
Username: hacluster
Password: ********
nodo1: Authorized
nodo2: Authorized
```

- En cada uno de los nodos del cluster, dentro del directorio /var/lib/pcsd, se crean:
  - Un fichero de identificación del nodo pcs_users.conf.
  - Un certificado de autenticación del nodo mediante clave pública en los ficheros pcsd.key y pcsd.crt.

Corosync (I)

- Utiliza paquetes multicast para la comunicación entre los nodos.
- Su fichero de configuración es /etc/corosync/corosync.conf.
  - Puede configurarse de forma manual.
  - Lo más comodo es crearlo de forma automática mediante el comando:

```
pcs cluster setup --name miCluster nodo1 nodo2
```

Fichero corosync.conf (I)

- Formado por líneas que configuran directivas:
  - Las directivas configuran el funcionamiento de corosync.
- Cinco tipo de directivas:
  - totem{}: Configuración de la comunicación del cluster.
  - logging{}: Configuración de los logs de la aplicación.
  - quorum{}: Configuración del número de nodos del cluster.
  - nodelist{}: Configuración de opciones de los nodos del cluster. Solo puede contener subdirectivas node{} para cada uno de los nodos miembros del cluster.
  - qb{}: Directivas de configuración de opciones de la librería libqb.

Fichero corosync.conf (II)

```
totem {
   version: 2
   secauth: off
   clustername: miCluster
   transport: udpu
}
nodelist {
   node {
      ring0_addr: nodo1
      nodeid: 1 }
   node {
      ring0_addr: nodo2
      nodeid: 2 }
}
quorum {
   provider: corosync_votequorum
}
logging {
   to_syslog: yes
}
```

Fichero corosync.conf (III)

- totem{} version:
  - Versión del fichero de configuración.
  - Único valor valido 2.
- totem{} secauth:
  - Autenticación de los nodos.
  - Valor on autenticación (defecto), valor off no.
- totem{} clustername:
  - Nombre del cluster.
- totem {} tranport:
  - Protocolo de transporte (UDP por defecto).

Fichero corosync.conf (IV)

- nodelist{} ring0_addr:
  - Dirección IP que utilizarán los nodos.
  - Se pueden definir distintos conjuntos de direcciones eligiendo ringX_addr con distintos valores de X.
- nodelist{} nodeid:
  - Valor de 32 bits que identifica el nodo en el cluster.
  - Debe ser distinto para cada nodo del cluster.
  - No puede ser cero, pues es un valor reservado.

Fichero corosync.conf (V)

- quorum{} provider:
  - Algoritmo a utilizar para comprobar los nodos activos del cluster.
  - Único valor permitido corosync_votequorum.
- logging{} to_syslog:
  - Enviar los mensajes al syslog del sistema (valor yes, valor por defecto) o no.

Arranque de corosync (I)

- Corosync puede arrancarse/pararse manualmente con el comando:

```
pcs cluster {start|stop} [--all]
```

- Una vez arrancado corosync podemos comprobar el estado de un nodo con:

```
corosync-cfgtool -s
```

```
Printing ring status.
Local node ID 1
RING ID 0
   id = 10.0.0.1
   status = ring 0 active with no faults
```

Arranque de corosync (II)

- Y el estado del cluster mediante el comando:

```
corosync-cmapctl
```

- Y los nodos con:

```
corosync-cmapctl | grep members
```

```
runtime.totem.pg.mrp.srp.members.1.config_version (u64) = 0
runtime.totem.pg.mrp.srp.members.1.ip (str) = r(0) ip(10.0.0.1)
runtime.totem.pg.mrp.srp.members.1.join_count (u32) = 1
runtime.totem.pg.mrp.srp.members.1.status (str) = joined
runtime.totem.pg.mrp.srp.members.2.config_version (u64) = 0
runtime.totem.pg.mrp.srp.members.2.ip (str) = r(0) ip(10.0.0.2)
runtime.totem.pg.mrp.srp.members.2.join_count (u32) = 1
runtime.totem.pg.mrp.srp.members.2.status (str) = joined
```

Arranque de corosync (III)

- Y el quorum (nodos) que forman el cluster con:

```
ccs status corosync
Membership information
----------------------
    Nodeid      Votes Name
         1          1 nodo1 (local)
         2          1 nodo2
```

Pacemaker (I)

- El fichero de configuración de pacemaker es /var/lib/pacemaker/cib/cib.xml.
  - Fichero muy complicado de editar de forma manual.
  - Deben utilizarse el comando pcs para su modificación.
- Inicialmente pacemaker conoce:
  - Nodos del cluster.
  - Aplicación que proporciona la infraestructura del cluster.
  - Versión de pacemaker.

Pacemaker (II)

```
> pcs config
  Cluster Name: miCluster
  Corosync Nodes:
     nodo1 nodo2
  Pacemaker Nodes:
     nodo1 nodo2

  Resources:

  Location Constraints:
  Ordering Constraints:
  Colocation Constraints:

  Cluster Properties:
   cluster-infrastructure: corosync
   dc-version: 1.1.11-1.fc19-9d39a6b
```

Pacemaker (III)

- La configuración inicial es inconsistente:
  - La configuración incluye el uso de stonith.
  - Stonith reinicia un nodo cuando cree que su estado es indeterminado (funciona mal).
  - Al no estar configurado stonith, se indica que la configuración es invalida.
- Debe deshabilitarse stonith si no esta configurado.

```
pcs property set stonith-enable=false
```

Pacemaker (IV)

- Existe una propiedad fundamental de un cluster que es el quorum.
  - Existe quorum cuando más de la mitad de los nodos de un cluster están online.
  - Si un cluster no tiene quorum, por defecto pacemaker detiene todos los servicios del cluster.
- En un cluster de dos nodos, si un nodo falla ya no se produce el quorum.
  - Pacemaker detendría todos los servicios.
  - No sería un cluster de alta disponibilidad.

Pacemaker (V)

- Se desactiva la detención de servicios en caso de que no haya quorum con el comando:

```
pcs property set no-quorum-policy=ignore
```

- Pacemaker intenta repartir los recursos entre todos los nodos disponibles.
- El reparto de los recursos en los nodos se configura con la propiedad:

```
rsc defaults resource-stickiness
```

Pacemaker (VI)

Valor	Descripción
0 (cero)	Los recursos se moverán de un nodo a otro en función de la carga en cada momento.
>0 (mayor que cero)	Los recursos permanecerán en el nodo actual, pero puede moverse a otro nodo cuando se considere necesario. Cuando más alto es el valor menor probabilidad de que algún recurso se mueva a otro nodo.
<0 (menor que cero)	Los recursos tienen preferencia a moverse a otro nodo cuando se considere necesario. Cuanto mayor valor absoluto, mayor probabilidad de que algún recurso se mueva a otro nodo.
INFINITY	Los recursos se quedarán en los nodos en los que se encuentran, y solo abandonaran los nodos ante un fallo en los mismos.
-INFINITY	Los recursos se moverán siempre de su ubicación actual.

Pacemaker (VII)

- El arranque automático de los recursos se configura con la propiedad:

  `symmetric-cluster`

  - Si su valor es true, entonces los recursos pueden arrancar en cualquier nodo.
  - Si el valor es false, entonces un recurso solo arranca en el nodo que se le indica explicitamente.
- En nuestro caso la configuración debe ser:

```
pcs resource rsc defaults resource-stickiness="INFINITY"
pcs property set symmetric-cluster="true"
```

Pacemaker (VIII)

- El estado actual de configuración del cluster es:

```
Cluster Name: miCluster
Corosync Nodes:
   nodo1 nodo2
Pacemaker Nodes:
   nodo1 nodo2
Resources:
Location Constraints:
Ordering Constraints:
Colocation Constraints:
Cluster Properties:
   cluster-infrastructure: corosync
   dc-version: 1.1.11-1.fc19-9d39a6b
   no-quorum-policy: ignore
   stonith-enabled: false
   symmetric-cluster: true
```

Pacemaker (IX)

- La configuración de cada uno de los recursos que ofrece el cluster se realiza como:

```
pcs resource create <resource> <[class:provider:]type> [resource options] [op <operation action> <operation options> [<operation action> <operation options>]...] [meta <meta options>...] [--clone|--master]
```

- Los únicos parámetros obligatorios son:
  - <recurso>: Nombre que tendrá el recurso que estamos configurando.
  - <type>: Tipo del recurso que estamos configurando, se corresponde con el nombre del script que inicia y detiene el recurso.

Pacemaker (X)

- <class> especifica la clase del recurso que estamos creando:
  - ocf: Open Cluster Framework, son scripts de inicio de forma que los valores devueltos, etc., corresponden a los esperados por pacemaker según el resultado de ejecución del script. Son los que deberían usarse si es posible.
  - lsb: Linux Standard Base, script de inicio de los servicios de linux en SystemV.
  - systemd: Scripts de inicio de los servicios de linux en Systemd.
  - service: Scripts de inicio de servicios, sean SystemV o systemd.
  - stonith: Usados para la comunicación con stonith.

Pacemaker (XI)

- Dentro de una clase pueden existir proveedores distintos que proporcionen recursos para la misma clase.
- El parámetro <provider> permite elegir el proveedor.
- Si se desean ver los recursos de una clase y proveedor:

```
pcs resource agents <clase>:<proveedor>
```

- Y si se desean obtener todas las clases y recursos existentes:

```
pcs resource list
```

Pacemaker (XII)

- El resto de opciones, etc., depende del script de configuración.
- Si se desea ver todas las opciones, etc., de un script se ejecuta:

```
pcs resource describe <class>:<proveedor>:<tipo>
```

- Por ejemplo:

```
>pcs resource describe ocf:heartbeat:IPaddr2

Resource options for: IPaddr2
ip (required): The IPv4 address to be configured in
dotted quad notation, for example"192.168.1.1".
nic: The base network interface on which the IP
address will be broughtonline.
...
```

Pacemaker (XIII)

- Ejemplos de configuración de recursos:
  - Dirección IP flotante:

```
pcs resource create ClusterIP ocf:heartbeat:IPaddr2 ip="10.0.0.3"
cidr_netmask=24
pcs resource add_operation ClusterIP monitor interval=30s
```

  - DRBD:

```
pcs resource create DrbdDisk ocf:redhat:drbd.sh resource="drbd0"
```

  - Montaje del DRBD:

```
pcs resource create fileSys ocf:heartbeat:Filesystem
device="/dev/drbd0" directory="/var/lib/mysql" fstype="ext4"
pcs resource add_operation fileSys start interval="0"
timeout="60s"
pcs resource add_operation fileSys stop interval="0"
timeout="60s"
```

Pacemaker (XIV)

- Ejemplos de configuración de recursos:
  - Servidor de MySQL:

```
pcs resource create Mysql ocf:heartbeat:mysql
binary="/usr/bin/mysqld_safe"
```

  - Servidor web Apache:

```
pcs resource create Httpd ocf:heartbeat:apache params
configfile="/etc/httpd/conf/httpd.conf" port="80"
pcs resource add_operation Httpd start interval="0"
pcs resource add_operation Httpd stop interval="0"
pcs resource add_operation Httpd monitor interval="5s"
timeout="20s"
```

Pacemaker (XV)

- En la configuración de Apache es necesario añadir:

```
PidFile run/httpd.pid

...
<Location /server-status>
    SetHandler server-status
    <RequireAll>
        Require all granted
        Require ip 127.0.0.1
    </RequireAll>
</Location>
```

Pacemaker (XVI)

- Faltan dos problemas por resolver:
  - Que todos los recursos se ejecuten en el mismo nodo.

```
pcs constraint colocation add <resource1> <resource2> <score>
```

  - Que los recursos se arranquen y paren en el orden correcto.

```
pcs constraint [action] <first rsc> then [action] <then rsc> [options]
```

Pacemaker (XVII)

- En nuestro caso:

```
pcs constraint colocation add ClusterIP DrbdDisk INFINITY
pcs constraint colocation add DrbdDisk fileSys INFINITY
pcs constraint colocacion add fileSys Mysql INFINITY
pcs constraint colocation add Mysql Httpd INFINITY

pcs constraint order ClusterIP then DrbdDisk
pcs constraint order DrbdDisk then fileSys
pcs constraint order fileSys then Mysql
pcs constraint order Mysql then Httpd
```

- La configuración final del cluster se puede ver con el comando:

```
pcs config
```

Funcionamiento del cluster (I)

- El funcionamiento del cluster se puede comprobar con los comandos:
  - pcs status: Muestra el estado del cluster.
  - pcs cluster unstandby <nodo>: Activa el nodo indicado.
  - pcs cluster standby <nodo>: Desactiva el nodo indicado.

Funcionamiento del cluster (II)

```
Cluster name: miCluster
Last updated: Thu Apr 10 22:03:20 2014
Last change: Thu Apr 10 22:02:20 2014 via crm_attribute on nodo1
Stack: corosync
Current DC: nodo1 (1) - partition with quorum
Version: 1.1.11-1.fc19-9d39a6b
2 Nodes configured
5 Resources configured


Node nodo1 (1): standby
Node nodo2 (2): standby

Full list of resources:

 fileSys(ocf::heartbeat:Filesystem):  Stopped
 Mysql  (ocf::heartbeat:mysql):   Stopped
 DrbdDisk   (ocf::redhat:drbd.sh):Stopped
 Httpd  (ocf::heartbeat:apache):  Stopped
 ClusterIP  (ocf::heartbeat:IPaddr2): Stopped
```

Funcionamiento del cluster (III)

```
Cluster name: miCluster
Last updated: Thu Apr 10 22:08:14 2014
Last change: Thu Apr 10 22:08:02 2014 via crm_attribute on nodo1
Stack: corosync
Current DC: nodo1 (1) - partition with quorum
Version: 1.1.11-1.fc19-9d39a6b
2 Nodes configured
5 Resources configured


Node nodo2 (2): standby
Online: [ nodo1 ]

Full list of resources:

 fileSys(ocf::heartbeat:Filesystem):  Started nodo1
 Mysql  (ocf::heartbeat:mysql):   Started nodo1
 DrbdDisk   (ocf::redhat:drbd.sh):Started nodo1
 Httpd  (ocf::heartbeat:apache):  Started nodo1
 ClusterIP  (ocf::heartbeat:IPaddr2): Started nodo1
```

Funcionamiento del cluster (IV)

```
Cluster name: miCluster
Last updated: Thu Apr 10 22:10:54 2014
Last change: Thu Apr 10 22:10:52 2014 via crm_attribute on nodo1
Stack: corosync
Current DC: nodo1 (1) - partition with quorum
Version: 1.1.11-1.fc19-9d39a6b
2 Nodes configured
5 Resources configured


Online: [ nodo1 nodo2 ]

Full list of resources:

 fileSys(ocf::heartbeat:Filesystem):  Started nodo1
 Mysql  (ocf::heartbeat:mysql):   Started nodo1
 DrbdDisk   (ocf::redhat:drbd.sh):Started nodo1
 Httpd  (ocf::heartbeat:apache):  Started nodo1
 ClusterIP  (ocf::heartbeat:IPaddr2): Started nodo1
```

Funcionamiento del cluster (V)

```
Cluster name: miCluster
Last updated: Thu Apr 10 22:13:40 2014
Last change: Thu Apr 10 22:13:25 2014 via crm_attribute on nodo1
Stack: corosync
Current DC: nodo1 (1) - partition with quorum
Version: 1.1.11-1.fc19-9d39a6b
2 Nodes configured
5 Resources configured


Node nodo1 (1): standby
Online: [ nodo2 ]

Full list of resources:

 fileSys(ocf::heartbeat:Filesystem):  Started nodo2
 Mysql  (ocf::heartbeat:mysql):   Started nodo2
 DrbdDisk   (ocf::redhat:drbd.sh):Started nodo2
 Httpd  (ocf::heartbeat:apache):  Started nodo2
 ClusterIP  (ocf::heartbeat:IPaddr2): Started nodo2
```

Funcionamiento del cluster (VI)

```
Cluster name: miCluster
Last updated: Thu Apr 10 22:19:54 2014
Last change: Thu Apr 10 22:15:57 2014 via crm_attribute on nodo1
Stack: corosync
Current DC: nodo1 (1) - partition WITHOUT quorum
Version: 1.1.11-1.fc19-9d39a6b
2 Nodes configured
5 Resources configured


Online: [ nodo1 ]
OFFLINE: [ nodo2 ]

Full list of resources:

 fileSys(ocf::heartbeat:Filesystem):  Started nodo1
 Mysql  (ocf::heartbeat:mysql):   Started nodo1
 DrbdDisk   (ocf::redhat:drbd.sh):Started nodo1
 Httpd  (ocf::heartbeat:apache):  Started nodo1
 ClusterIP  (ocf::heartbeat:IPaddr2): Started nodo1
```