

Dernière mise-à-jour : 2021/01/26 14:18

HAR200 - Gestion de la Haute Disponibilité avec OpenSVC sous RHEL 7

Contenu du Module

- **HAR200 - Gestion de la Haute Disponibilité avec OpenSVC sous CentOS 7**

- Contenu du Module
 - Présentation
 - L'Agent
 - Collector
 - Mise en Place
 - Préparation des Machines Virtuelles
 - Configuration du node1.opensvc.loc
 - Configuration du node2.opensvc.loc
 - Mise en place des LVM de test
 - Configuration du node1.opensvc.loc ET node2.opensvc.loc
 - Installation d'opensvc-agent
 - Installation sur node1.opensvc.loc et node2.opensvc.loc
 - Clefs SSH
 - Création d'un Service
 - Créer le fichier de service
 - Le Répertoire des Scripts de Démarrage des Services
 - Service Management Facility
 - Intégrer une Application
 - Applications launcher directory
 - Obtenir le binaire de l'application
 - Service Failover
 - Haute Disponibilité
-

Présentation

Le projet **OpenSVC** est né au mois de novembre 2009. OpenSVC comprend deux composants logiciels :

L'Agent

L'agent OpenSVC est un Gestionnaire de Ressources du Cluster publié sous la licence **GNU General Public License v2** ayant des fonctions de reporting et de gestion de la configuration. Couplé avec un logiciel **heartbeat**, l'agent est un outil de gestion d'un cluster complet.

Plus d'information concernant l'agent est disponible [ici](#).

Collector

Le **Collector** est un logiciel **commercial** optionnel, compatible web 2.0. Comme son nom indique, il collecte des informations à partir des agents ainsi qu'à partir des éléments de l'infrastructure du site tel le réseau, les SANs, les serveurs etc.)

Mise en Place

Préparation des Machines Virtuelles

A partir de votre machine virtuelle **CentOS 7**, créez 2 clones complets :

```
VBoxManage clonevm CentOS_7 --name="node1.i2tch.loc" --register --mode=all  
VBoxManage clonevm CentOS_7 --name="node2.i2tch.loc" --register --mode=all
```

Modifiez la configuration réseau des deux clones :

Adaptateur	Carte 1	Carte 2	Carte 3
Type de réseau	NAT	intnet	intnet



Important - Dans Virtual Box > Paramètres de node2.opensvc.loc > Réseau > Carte 1 > Redirection de ports, Modifiez le port hôte ssh en 4022.

Démarrez les machines virtuelles **node1.opensvc.loc** et **node2.opensvc.loc** et modifiez les noms d'hôtes ainsi :

```
[root@centos7 ~]# nmcli general hostname node1.opensvc.loc
[root@centos7 ~]# hostname
node1.opensvc.loc
```

```
[root@centos7 ~]# nmcli general hostname node2.opensvc.loc
[root@centos7 ~]# hostname
node2.opensvc.loc
```



Important - Déconnectez-vous de et re-connectez-vous à chaque VM.

Vérifiez la configuration réseau sur chaque noeud :

```
[root@node1 ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:35:2a:bc brd ff:ff:ff:ff:ff:ff
```

```
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 86321sec preferred_lft 86321sec
    inet6 fe80::9e30:f271:f425:faa2/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:9a:f3:96 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::c141:816b:e364:4dc/64 scope link
        valid_lft forever preferred_lft forever
4: enp0s9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:84:d3:59 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::e1a1:e92e:b0ab:3b40/64 scope link
        valid_lft forever preferred_lft forever
```

```
[root@node2 ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:90:fe:5a brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 86306sec preferred_lft 86306sec
    inet6 fe80::4c50:7b9a:e166:b0a9/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:39:0b:1d brd ff:ff:ff:ff:ff:ff
    inet6 fe80::1ccc:8e24:f0b6:8de2/64 scope link
        valid_lft forever preferred_lft forever
4: enp0s9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:66:86:7f brd ff:ff:ff:ff:ff:ff
    inet6 fe80::bb82:6fa3:a503:b275/64 scope link
        valid_lft forever preferred_lft forever
```

Ethernet Channel Bonding

Le **Channel Bonding** est un regroupement d'interfaces réseau sur le même serveur afin de mettre en place la redondance ou d'augmenter les performances.

Le Channel Bonding est géré nativement sous Linux. Aucune application tierce n'est requise.

Configuration du node1.opensvc.loc

Assurez-vous que le module **bonding** soit chargé :

```
[root@node1 ~]# lsmod | grep bonding
[root@node1 ~]# modprobe bonding
[root@node1 ~]# lsmod | grep bonding
bonding                145728  0
```

Consultez la configuration des interfaces réseaux :

```
[root@node1 ~]# nmcli c show
```

NAME	UUID	TYPE	DEVICE
Wired connection 1	b24208cf-9b3d-3b5f-b9ea-95668778104f	802-3-ethernet	enp0s3
Wired connection 2	d3aa4096-a44f-39e2-adb8-a59c542ed02b	802-3-ethernet	--
Wired connection 3	d9159b5d-0d16-3930-9662-b0c1b90f2450	802-3-ethernet	--

Créez le fichier **/etc/sysconfig/network-scripts/ifcfg-bond0** :

```
[root@node1 ~]# vi /etc/sysconfig/network-scripts/ifcfg-bond0
[root@node1 ~]# cat /etc/sysconfig/network-scripts/ifcfg-bond0
DEVICE=bond0
USERCTL=no
BOOTPROTO=None
ONBOOT=yes
```

```
IPADDR=192.168.121.1
NETMASK=255.255.255.0
NETWORK=192.168.121.0
BONDING_OPTS="miimon=100 mode=balance-xor"
TYPE=Unknown
IPV6INIT=no
```

Créez le fichier **/etc/sysconfig/network-scripts/ifcfg-enp0s8** :

```
[root@node1 ~]# vi /etc/sysconfig/network-scripts/ifcfg-enp0s8
[root@node1 ~]# cat /etc/sysconfig/network-scripts/ifcfg-enp0s8
DEVICE=enp0s8
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
USERCTL=no
```

Créez le fichier **/etc/sysconfig/network-scripts/ifcfg-enp0s9** :

```
[root@node1 ~]# vi /etc/sysconfig/network-scripts/ifcfg-enp0s9
[root@node1 ~]# cat /etc/sysconfig/network-scripts/ifcfg-enp0s9
DEVICE=enp0s9
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
USERCTL=no
```

Créez le fichier **/etc/modprobe.d/bonding.conf** :

```
[root@node1 ~]# vi /etc/modprobe.d/bonding.conf
[root@node1 ~]# cat /etc/modprobe.d/bonding.conf
```

```
alias bond0 bonding
```

Modifiez le fichier **/etc/hosts** :

```
[root@node1 ~]# vi /etc/hosts
[root@node1 ~]# cat /etc/hosts
127.0.0.1      localhost.localdomain localhost
::1          localhost6.localdomain6 localhost6
192.168.121.1  node1.opensvc.loc      node1
192.168.121.2  node2.opensvc.loc      node2
192.168.121.42 svc1.opensvc.loc   svc1
```

Re-démarrez le service **network** :

```
[root@node1 ~]# nmcli c show
```

NAME	UUID	TYPE	DEVICE
System enp0s8	00cb8299-feb9-55b6-a378-3fdc720e0bc6	802-3-ethernet	enp0s8
System enp0s9	93d13955-e9e2-a6bd-df73-12e3c747f122	802-3-ethernet	enp0s9
Wired connection 1	4ea414aa-a08a-3a0d-8d8c-09aca9bd7fcc	802-3-ethernet	enp0s3
bond0	4e7fb7b3-d782-4952-b7dc-bc26cf1f5e9a	bond	bond0
Wired connection 2	d6c9febc-845b-3e4b-91eb-a9f001670cdc	802-3-ethernet	--
Wired connection 3	b9978cef-7b73-318f-aeeb-454d9a1ffeeb	802-3-ethernet	--

Vérifiez la configuration du réseau :

```
[root@node1 ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 08:00:27:41:5b:d7 brd ff:ff:ff:ff:ff:ff
```

```
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 85360sec preferred_lft 85360sec
    inet6 fe80::647b:5dad:397d:3ebb/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast master bond0 state UP qlen 1000
    link/ether 08:00:27:59:5a:01 brd ff:ff:ff:ff:ff:ff
4: enp0s9: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast master bond0 state UP qlen 1000
    link/ether 08:00:27:59:5a:01 brd ff:ff:ff:ff:ff:ff
5: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP qlen 1000
    link/ether 08:00:27:59:5a:01 brd ff:ff:ff:ff:ff:ff
    inet 192.168.121.1/24 brd 192.168.121.255 scope global bond0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe59:5a01/64 scope link tentative dadfailed
        valid_lft forever preferred_lft forever
```

Consultez la configuration de l'interface **bond0** :

```
[root@node1 ~]# cat /proc/net/bonding/bond0
Ethernet Channel Bonding Driver: v3.7.1 (April 27, 2011)

Bonding Mode: load balancing (xor)
Transmit Hash Policy: layer2 (0)
MII Status: up
MII Polling Interval (ms): 100
Up Delay (ms): 0
Down Delay (ms): 0

Slave Interface: enp0s8
MII Status: up
Speed: 1000 Mbps
Duplex: full
Link Failure Count: 0
Permanent HW addr: 08:00:27:59:5a:01
Slave queue ID: 0
```

```
Slave Interface: enp0s9
MII Status: up
Speed: 1000 Mbps
Duplex: full
Link Failure Count: 0
Permanent HW addr: 08:00:27:43:66:3e
Slave queue ID: 0
```

Configuration du node2.opensvc.loc

Assurez-vous que le module **bonding** soit chargé :

```
[root@node2 ~]# lsmod | grep bonding
[root@node2 ~]# modprobe bonding
[root@node2 ~]# lsmod | grep bonding
bonding                145728  0
```

Consultez la configuration des interfaces réseaux :

```
[root@node2 ~]# nmcli c show
```

NAME	UUID	TYPE	DEVICE
Wired connection 1	794e172c-62ad-3e46-9ee4-ac8f28a250e3	802-3-ethernet	enp0s3
Wired connection 2	6f91b884-972c-340c-a61c-7a2a67305ad9	802-3-ethernet	enp0s8
Wired connection 3	4b7cbfcd-0558-3000-983b-39db9485f7ae	802-3-ethernet	enp0s9

Créez le fichier **/etc/sysconfig/network-scripts/ifcfg-bond0** :

```
[root@node2 ~]# vi /etc/sysconfig/network-scripts/ifcfg-bond0
[root@node2 ~]# cat /etc/sysconfig/network-scripts/ifcfg-bond0
DEVICE=bond0
USERCTL=no
BOOTPROTO=None
```

```
ONBOOT=yes
IPADDR=192.168.121.2
NETMASK=255.255.255.0
NETWORK=192.168.121.0
BONDING_OPTS="miimon=100 mode=balance-xor"
TYPE=Unknown
IPV6INIT=no
```

Créez le fichier **/etc/sysconfig/network-scripts/ifcfg-enp0s8** :

```
[root@node2 ~]# vi /etc/sysconfig/network-scripts/ifcfg-enp0s8
[root@node2 ~]# cat /etc/sysconfig/network-scripts/ifcfg-enp0s8
DEVICE=enp0s8
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
USERCTL=no
```

Créez le fichier **/etc/sysconfig/network-scripts/ifcfg-enp0s9** :

```
[root@node2 ~]# vi /etc/sysconfig/network-scripts/ifcfg-enp0s9
[root@node2 ~]# cat /etc/sysconfig/network-scripts/ifcfg-enp0s9
DEVICE=enp0s9
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
USERCTL=no
```

Créez le fichier **/etc/modprobe.d/bonding.conf** :

```
[root@node2 ~]# vi /etc/modprobe.d/bonding.conf
[root@node2 ~]# cat /etc/modprobe.d/bonding.conf
```

```
alias bond0 bonding
```

Modifiez le fichier **/etc/hosts** :

```
[root@node2 ~]# vi /etc/hosts
[root@node2 ~]# cat /etc/hosts
127.0.0.1      localhost.localdomain localhost
::1          localhost6.localdomain6 localhost6
192.168.121.1  node1.opensvc.loc      node1
192.168.121.2  node2.opensvc.loc      node2
192.168.121.42 svc1.opensvc.loc      svc1
```

Re-démarrez le service **network** :

```
[root@node2 ~]# systemctl restart network
[root@node2 ~]# nmcli c show
```

NAME	UUID	TYPE	DEVICE
System enp0s8	00cb8299-feb9-55b6-a378-3fdc720e0bc6	802-3-ethernet	enp0s8
System enp0s9	93d13955-e9e2-a6bd-df73-12e3c747f122	802-3-ethernet	enp0s9
Wired connection 1	794e172c-62ad-3e46-9ee4-ac8f28a250e3	802-3-ethernet	enp0s3
bond0	8ee04199-caad-493f-80b3-003b16bf640e	bond	bond0
Wired connection 2	6f91b884-972c-340c-a61c-7a2a67305ad9	802-3-ethernet	--
Wired connection 3	4b7cbfcd-0558-3000-983b-39db9485f7ae	802-3-ethernet	--

Vérifiez la configuration du réseau :

```
[root@node2 ~]# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
```

```
link/ether 08:00:27:13:13:33 brd ff:ff:ff:ff:ff:ff
inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
    valid_lft 84965sec preferred_lft 84965sec
inet6 fe80::2761:4ee8:d2de:d0e2/64 scope link
    valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast master bond0 state UP qlen 1000
    link/ether 08:00:27:47:75:77 brd ff:ff:ff:ff:ff:ff
4: enp0s9: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast master bond0 state UP qlen 1000
    link/ether 08:00:27:47:75:77 brd ff:ff:ff:ff:ff:ff
5: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP qlen 1000
    link/ether 08:00:27:47:75:77 brd ff:ff:ff:ff:ff:ff
    inet 192.168.121.2/24 brd 192.168.121.255 scope global bond0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe47:7577/64 scope link
        valid_lft forever preferred_lft forever
```

Consultez la configuration de l'interface **bond0** :

```
[root@node2 ~]# cat /proc/net/bonding/bond0
Ethernet Channel Bonding Driver: v3.7.1 (April 27, 2011)

Bonding Mode: load balancing (xor)
Transmit Hash Policy: layer2 (0)
MII Status: up
MII Polling Interval (ms): 100
Up Delay (ms): 0
Down Delay (ms): 0

Slave Interface: enp0s9
MII Status: up
Speed: 1000 Mbps
Duplex: full
Link Failure Count: 0
Permanent HW addr: 08:00:27:47:75:77
```

```
Slave queue ID: 0
```

```
Slave Interface: enp0s8
```

```
MII Status: up
```

```
Speed: 1000 Mbps
```

```
Duplex: full
```

```
Link Failure Count: 0
```

```
Permanent HW addr: 08:00:27:5a:91:89
```

```
Slave queue ID: 0
```

Mise en place des LVM de test

Configuration du node1.opensvc.loc ET node2.opensvc.loc

Créez un fichier spécial de type bloc (b) ayant un majeur de 7 et un mineur de 0 et les permissions (-m) de 660 :

```
[root@node1 ~]# mknod -m 0660 /dev/loop0 b 7 0
```

```
[root@node2 ~]# mknod -m 0660 /dev/loop0 b 7 0
```

Associez le périphérique **/dev/loop0** avec le fichier spécial **/dev/sda1** :

```
[root@node1 ~]# losetup /dev/loop0 /dev/sda1
```

```
[root@node2 ~]# losetup /dev/loop0 /dev/sda1
```

Initialisez le LVM avec la commande **vgscan** :

```
[root@node1 ~]# vgscan  
Reading volume groups from cache.
```

```
[root@node2 ~]# vgscan
Reading volume groups from cache.
```

Créez ensuite un PV sur **/dev/loop0** :

```
[root@node1 ~]# pvcreate /dev/loop0
WARNING: xfs signature detected on /dev/loop0 at offset 0. Wipe it? [y/n]: n
Aborted wiping of xfs.
1 existing signature left on the device.
[root@node1 ~]#
[root@node1 ~]# pvcreate /dev/loop0
WARNING: xfs signature detected on /dev/loop0 at offset 0. Wipe it? [y/n]: y
Wiping xfs signature on /dev/loop0.
Physical volume "/dev/loop0" successfully created.
```

```
[root@node2 ~]# pvcreate /dev/loop0
WARNING: xfs signature detected on /dev/loop0 at offset 0. Wipe it? [y/n]: y
Wiping xfs signature on /dev/loop0.
Physical volume "/dev/loop0" successfully created.
```

Créez un LV dénommé **vgsvc1** sur **/dev/loop0** :

```
[root@node1 ~]# vgcreate -s 4M vgsvc1 /dev/loop0
Volume group "vgsvc1" successfully created
[root@node1 ~]#
[root@node1 ~]# vgdisplay
--- Volume group ---
VG Name                vgsvc1
System ID
Format                 lvm2
Metadata Areas         1
Metadata Sequence No   1
VG Access              read/write
```

VG Status	resizable
MAX LV	0
Cur LV	0
Open LV	0
Max PV	0
Cur PV	1
Act PV	1
VG Size	196.00 MiB
PE Size	4.00 MiB
Total PE	49
Alloc PE / Size	0 / 0
Free PE / Size	49 / 196.00 MiB
VG UUID	hhvl4f-avgL-2a5z-iT4G-9fMm-KpdC-xXz31C

```
[root@node2 ~]# vgcreate -s 4M vgsvc1 /dev/loop0
```

```
Volume group "vgsvc1" successfully created
```

```
[root@node2 ~]#
```

```
[root@node2 ~]# vgdisplay
```

```
--- Volume group ---
```

VG Name	vgsvc1
System ID	
Format	lvm2
Metadata Areas	1
Metadata Sequence No	1
VG Access	read/write
VG Status	resizable
MAX LV	0
Cur LV	0
Open LV	0
Max PV	0
Cur PV	1
Act PV	1
VG Size	196.00 MiB
PE Size	4.00 MiB

Total PE	49
Alloc PE / Size	0 / 0
Free PE / Size	49 / 196.00 MiB
VG UUID	hhvl4f-avgL-2a5z-iT4G-9fMm-KpdC-xXz31C

Créez deux LV de 96 Mo dénommés **lvappsvc1** et **lvdatasvc1** respectivement :

```
[root@node1 ~]# lvcreate -L 95 -n lvappsvc1 vgsvc1
Rounding up size to full physical extent 96.00 MiB
Logical volume "lvappsvc1" created.
[root@node1 ~]#
[root@node1 ~]# lvcreate -L 95 -n lvdatasvc1 vgsvc1
Rounding up size to full physical extent 96.00 MiB
Logical volume "lvdatasvc1" created.
[root@node1 ~]#
[root@node1 ~]# lvdisplay
--- Logical volume ---
LV Path                /dev/vgsvc1/lvappsvc1
LV Name                 lvappsvc1
VG Name                 vgsvc1
LV UUID                 LiIKMj-eE9N-dfxu-rjv1-F98Z-8Pr0-xCDd05
LV Write Access         read/write
LV Creation host, time  node1.opensvc.loc, 2018-09-12 07:28:47 +0200
LV Status                available
# open                  0
LV Size                 96.00 MiB
Current LE              24
Segments                1
Allocation              inherit
Read ahead sectors      auto
- currently set to     8192
Block device            253:0
--- Logical volume ---
LV Path                /dev/vgsvc1/lvdatasvc1
```

```
LV Name          lvdatasvc1
VG Name          vgsvc1
LV UUID          pisl34-3MJP-0ZxT-y7i0-FKG9-qbxG-7doXsS
LV Write Access   read/write
LV Creation host, time node1.opensvc.loc, 2018-09-12 07:29:54 +0200
LV Status        available
# open           0
LV Size          96.00 MiB
Current LE       24
Segments         1
Allocation        inherit
Read ahead sectors auto
- currently set to 8192
Block device     253:1
```

```
[root@node2 ~]# lvcreate -L 95 -n lvappsvc1 vgsvc1
Rounding up size to full physical extent 96.00 MiB
Logical volume "lvappsvc1" created.
[root@node2 ~]#
[root@node2 ~]# lvcreate -L 95 -n lvdatasvc1 vgsvc1
Rounding up size to full physical extent 96.00 MiB
Logical volume "lvdatasvc1" created.
[root@node2 ~]#
[root@node2 ~]# lvdisplay
--- Logical volume ---
LV Path          /dev/vgsvc1/lvappsvc1
LV Name          lvappsvc1
VG Name          vgsvc1
LV UUID          LiIKMj-eE9N-dfxu-rjv1-F98Z-8Pr0-xCDd05
LV Write Access   read/write
LV Creation host, time node2.opensvc.loc, 2018-09-12 07:28:47 +0200
LV Status        available
# open           0
LV Size          96.00 MiB
```

```
Current LE          24
Segments            1
Allocation           inherit
Read ahead sectors  auto
- currently set to  8192
Block device         253:0
--- Logical volume ---
LV Path              /dev/vgsvc1/lvdatasvc1
LV Name              lvdatasvc1
VG Name              vgsvc1
LV UUID              pisl34-3MJP-0ZxT-y7i0-FKG9-qbxG-7doXsS
LV Write Access      read/write
LV Creation host, time node2.opensvc.loc, 2018-09-12 07:29:54 +0200
LV Status            available
# open              0
LV Size              96.00 MiB
Current LE          24
Segments            1
Allocation           inherit
Read ahead sectors  auto
- currently set to  8192
Block device         253:1
```

Créez un système de fichiers **ext4** sur chaque LV :

```
[root@node1 ~]# mkfs.ext4 /dev/vgsvc1/lvappsvc1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
Stride=0 blocks, Stripe width=0 blocks
24576 inodes, 98304 blocks
4915 blocks (5.00%) reserved for the super user
```

```
First data block=1
Maximum filesystem blocks=33685504
12 block groups
8192 blocks per group, 8192 fragments per group
2048 inodes per group
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done
[root@node1 ~]#
[root@node1 ~]# mkfs.ext4 /dev/vgsvcl/lvdatasvc1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
Stride=0 blocks, Stripe width=0 blocks
24576 inodes, 98304 blocks
4915 blocks (5.00%) reserved for the super user
First data block=1
Maximum filesystem blocks=33685504
12 block groups
8192 blocks per group, 8192 fragments per group
2048 inodes per group
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done
```

```
[root@node2 ~]# mkfs.ext4 /dev/vgsvc1/lvappsvc1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
Stride=0 blocks, Stripe width=0 blocks
24576 inodes, 98304 blocks
4915 blocks (5.00%) reserved for the super user
First data block=1
Maximum filesystem blocks=33685504
12 block groups
8192 blocks per group, 8192 fragments per group
2048 inodes per group
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done
[root@node2 ~]#
[root@node2 ~]# mkfs.ext4 /dev/vgsvc1/lvdatasvc1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
Stride=0 blocks, Stripe width=0 blocks
24576 inodes, 98304 blocks
4915 blocks (5.00%) reserved for the super user
First data block=1
Maximum filesystem blocks=33685504
12 block groups
```

```
8192 blocks per group, 8192 fragments per group
2048 inodes per group
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done
```

Créez les points de montage **/svc1/data** et **/svc1/app** :

```
[root@node1 ~]# mkdir -p /svc1/data
[root@node1 ~]# mkdir /svc1/app
```

```
[root@node2 ~]# mkdir -p /svc1/data
[root@node2 ~]# mkdir /svc1/app
```

Dernièrement, montez les deux systèmes de fichiers :

```
[root@node1 ~]# mount -t ext4 /dev/vgsvc1/lvappsvc1 /svc1/app
[root@node1 ~]#
[root@node1 ~]# mount -t ext4 /dev/vgsvc1/lvdatasvc1 /svc1/data
```

```
[root@node2 ~]# mount -t ext4 /dev/vgsvc1/lvappsvc1 /svc1/app
[root@node2 ~]#
[root@node2 ~]# mount -t ext4 /dev/vgsvc1/lvdatasvc1 /svc1/data
```

Installation d'opensvc-agent

Installation sur node1.opensvc.loc et node2.opensvc.loc

Téléchargez la dernière version d'**opensvc-agent** :

```
[root@node1 ~]# wget -O /tmp/opensvc.latest.rpm https://repo.opensvc.com/rpms/current
--2018-09-12 05:01:14-- https://repo.opensvc.com/rpms/current
Resolving repo.opensvc.com (repo.opensvc.com)... 37.59.71.9
Connecting to repo.opensvc.com (repo.opensvc.com)|37.59.71.9|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1248463 (1.2M)
Saving to: '/tmp/opensvc.latest.rpm'

100%[=====>] 1,248,463 2.86MB/s in 0.4s

2018-09-12 05:01:15 (2.86 MB/s) - '/tmp/opensvc.latest.rpm' saved [1248463/1248463]
```

```
[root@node2 ~]# wget -O /tmp/opensvc.latest.rpm https://repo.opensvc.com/rpms/current
--2018-09-12 05:02:13-- https://repo.opensvc.com/rpms/current
Resolving repo.opensvc.com (repo.opensvc.com)... 37.59.71.9
Connecting to repo.opensvc.com (repo.opensvc.com)|37.59.71.9|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1248463 (1.2M)
Saving to: '/tmp/opensvc.latest.rpm'

100%[=====>] 1,248,463 296KB/s in 4.1s

2018-09-12 05:02:17 (296 KB/s) - '/tmp/opensvc.latest.rpm' saved [1248463/1248463]
```

Installez l'**opensvc-agent** :

```
[root@node1 ~]# yum -y install /tmp/opensvc.latest.rpm
```

```
[root@node2 ~]# yum -y install /tmp/opensvc.latest.rpm
```

Démarrez et activez le service **opensvc-agent** :

```
[root@node1 ~]# systemctl start opensvc-agent.service
[root@node1 ~]# systemctl is-active opensvc-agent.service
active
```

```
[root@node2 ~]# systemctl start opensvc-agent.service
[root@node2 ~]# systemctl is-active opensvc-agent.service
active
```

Vérifiez le statut du service **opensvc-agent** :

```
[root@node1 ~]# systemctl status opensvc-agent
● opensvc-agent.service - OpenSVC agent
   Loaded: loaded (/usr/lib/systemd/system/opensvc-agent.service; enabled; vendor preset: disabled)
   Active: active (running) since Wed 2018-09-12 05:03:30 CEST; 5min ago
     Docs: http://docs.opensvc.com/
           file:/usr/share/doc/opensvc/
           man:nodemgr(1)
           man:svcmgr(1)
           man:svcmmon(1)
  Process: 7532 ExecStart=/usr/share/opensvc/bin/nodemgr daemon start (code=exited, status=0/SUCCESS)
 Main PID: 7555 (python)
    CGroup: /system.slice/opensvc-agent.service
            └─7555 /usr/bin/python /usr/share/opensvc/lib/osvcd.py

Sep 12 05:03:30 node1.opensvc.loc python[7555]: node1.opensvc.loc.osvcd.moni...e
Sep 12 05:03:30 node1.opensvc.loc python[7555]: node1.opensvc.loc.osvcd.moni...h
Sep 12 05:03:31 node1.opensvc.loc python[7555]: node1.opensvc.loc.osvcd.moni...n
Sep 12 05:03:31 node1.opensvc.loc python[7555]: node1.opensvc.loc.osvcd.moni...r
Sep 12 05:03:31 node1.opensvc.loc python[7555]: node1.opensvc.loc.osvcd.moni...e
Sep 12 05:03:31 node1.opensvc.loc python[7555]: node1.opensvc.loc.osvcd.moni...e
Sep 12 05:03:32 node1.opensvc.loc python[7555]: node1.opensvc.loc.osvcd node...)
Sep 12 05:03:32 node1.opensvc.loc python[7555]: node1.opensvc.loc.osvcd.moni...e
```

```
Sep 12 05:03:33 node1.opensvc.loc python[7555]: node1.opensvc.loc.osvcd.list...e
Sep 12 05:04:34 node1.opensvc.loc python[7555]: node1.opensvc.loc.osvcd.sche...5
Hint: Some lines were ellipsized, use -l to show in full.
```

```
[root@node2 ~]# systemctl status opensvc-agent
● opensvc-agent.service - OpenSVC agent
   Loaded: loaded (/usr/lib/systemd/system/opensvc-agent.service; enabled; vendor preset: disabled)
   Active: active (running) since Wed 2018-09-12 05:03:53 CEST; 3min 29s ago
     Docs: http://docs.opensvc.com/
           file:/usr/share/doc/opensvc/
           man:nodemgr(1)
           man:svcmgr(1)
           man:svcmmon(1)
  Process: 7655 ExecStart=/usr/share/opensvc/bin/nodemgr daemon start (code=exited, status=0/SUCCESS)
 Main PID: 7687 (python)
    CGroup: /system.slice/opensvc-agent.service
            └─7687 /usr/bin/python /usr/share/opensvc/lib/osvcd.py
```

```
Sep 12 05:03:53 node2.opensvc.loc python[7687]: node2.opensvc.loc.osvcd.moni...h
Sep 12 05:03:54 node2.opensvc.loc python[7687]: node2.opensvc.loc.osvcd.moni...n
Sep 12 05:03:54 node2.opensvc.loc python[7687]: node2.opensvc.loc.osvcd.moni...r
Sep 12 05:03:54 node2.opensvc.loc python[7687]: node2.opensvc.loc.osvcd.moni...e
Sep 12 05:03:54 node2.opensvc.loc python[7687]: node2.opensvc.loc.osvcd.moni...e
Sep 12 05:03:55 node2.opensvc.loc python[7687]: node2.opensvc.loc.osvcd node...)
Sep 12 05:03:55 node2.opensvc.loc python[7687]: node2.opensvc.loc.osvcd.moni...e
Sep 12 05:03:56 node2.opensvc.loc python[7687]: node2.opensvc.loc.osvcd.list...e
Sep 12 05:04:56 node2.opensvc.loc python[7687]: node2.opensvc.loc.osvcd.sche...3
Sep 12 05:06:46 node2.opensvc.loc python[7687]: node2.opensvc.loc.osvcd.sche...8
Hint: Some lines were ellipsized, use -l to show in full.
```

Utilisez la commande **svcmmon** pour visualiser le statut des services sur le noeud :

```
[root@node1 ~]# svcmmon
```

```
Threads                                node1.opensvc.loc
listener running 0.0.0.0:1214
monitor running
scheduler running 6 wait

Nodes                                node1.opensvc.loc
score                                | 92
load 15m                            | 0.1
mem                                  | 50/98%:488m
swap                                | 1/90%:1g
state                                |

Services                             node1.opensvc.loc
```

```
[root@node2 ~]# svcmon
```

```
Threads                                node2.opensvc.loc
listener running 0.0.0.0:1214
monitor running
scheduler running 6 wait

Nodes                                node2.opensvc.loc
score                                | 92
load 15m                            | 0.1
mem                                  | 50/98%:488m
swap                                | 1/90%:1g
state                                |

Services                             node2.opensvc.loc
```

L'agent utilise des threads pour gérer des rôles différents :

- **Listener** - à l'écoute de requêtes des daemon d'autres neouds, du collector, de nodemgr et de svcmgr,
- **Monitor** - gère les actions globales de démarrage du cluster et de failover,

- **Scheduler** - gère la gestion des tâches sur les neouds,
- **Heartbeat** - vérifie si les autres neouds sont “en vie” et échange des informations entre les neouds.

Pour mettre en place un heartbeat, saisissez la commande suivante :

```
[root@node1 ~]# nodemgr set --param hb#1.type --value unicast
[root@node1 ~]# svcmon
```

```
Threads                                node1.opensvc.loc
hb#1.rx    running 0.0.0.0:10000 | /
hb#1.tx    running                | /
listener   running 0.0.0.0:1214
monitor    running
scheduler  running 5 wait
```

```
Nodes                                node1.opensvc.loc
score                                           | 92
load 15m                                       | 0.1
mem                                           | 51/98%:488m
swap                                          | 1/90%:1g
state                                         |
```

```
Services                                node1.opensvc.loc
```

La configuration de chaque noeud est stockée dans le fichier **/etc/opensvc/node.conf**.

```
[root@node1 ~]# cat /etc/opensvc/node.conf
[cluster]
id = 6de28056-b638-11e8-bb2d-080027415bd7
nodes = node1.opensvc.loc
name = default
secret = 6e828240b63811e8819b080027415bd7

[hb#1]
```

```
type = unicast
```

Les deux nodes sont en mode **TST** par défaut. En production, le mode des noeuds doit être modifié à **PRD** de la façon suivante :

```
# nodemgr set --param node.env --value PRD
```

Clefs SSH

Les membres du cluster ont besoin d'effectuer des authentifications mutuelles par ssh afin d'échanger des fichiers de configuration. Chaque neoud utilisera une authentification par clefs :

- node1 se connectera à node2 en tant que root,
- node2 se connectera à node1 en tant que root.

Transférez donc la clef de node1 à node2 :

```
[root@node1 ~]# ssh-copy-id root@node2
/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host 'node2 (192.168.121.2)' can't be established.
ECDSA key fingerprint is SHA256:Rg0sp/XI7JHNq+oIfHKw+jkHdtTnBIh+Dd7kVmHRxtU.
ECDSA key fingerprint is MD5:19:cd:05:58:af:2c:10:82:52:ba:e3:31:df:bd:72:54.
Are you sure you want to continue connecting (yes/no)? yes
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@node2's password: fenestros
```

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@node2'"
and check to make sure that only the key(s) you wanted were added.

```
[root@node1 ~]# ssh node2 hostname
```

```
node2.opensvc.loc
```

Transférez donc la clef de node2 à node1 :

```
[root@node2 ~]# ssh-copy-id root@node1
/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host 'node1 (192.168.121.1)' can't be established.
ECDSA key fingerprint is SHA256:Rg0sp/XI7JHNq+oIfHKw+jkHdtTnBIh+Dd7kVmHRxtU.
ECDSA key fingerprint is MD5:19:cd:05:58:af:2c:10:82:52:ba:e3:31:df:bd:72:54.
Are you sure you want to continue connecting (yes/no)? yes
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@node1's password: fenestros

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'root@node1'"
and check to make sure that only the key(s) you wanted were added.

[root@node2 ~]# ssh node1 hostname
node1.opensvc.loc
```

Création d'un Service

Créez un service dénommé **svc1** :

```
[root@node1 ~]# svcmgr -s svc1 create
```

Cette commande génère un fichier dans **/etc/opensvc** appelé **svc1.conf** :

```
[root@node1 ~]# cat /etc/opensvc/svc1.conf
[DEFAULT]
```

```
id = 7adc8d25-1668-4078-a3f0-5b4472c436d2
```

Ce fichier est vide. Il convient donc de l'éditer afin de définir le service.

Créer le fichier de service

Editez donc ce fichier afin de créer un service appelé **MyApp** qui s'exécute sur node1 avec un failover vers node2. Ce service utilisera une adresse IP dénommée svc1.opensvc.loc, un VG dénommé **vgsvc1** et deux systèmes de fichiers sur les LV **/dev/mapper/vgsvc1-lvappsvc1** et **/dev/mapper/vgsvc1-lvdatasvc1** :

```
[root@node1 ~]# svcmgr -s svc1 edit config
[root@node1 ~]# cat /etc/opensvc/svc1.conf
[DEFAULT]
id = 7adc8d25-1668-4078-a3f0-5b4472c436d2
app = MyApp
nodes = node1.opensvc.loc node2.opensvc.loc

[ip#0]
ipname = svc1.opensvc.loc
ipdev = bond0

[disk#0]
type = vg
name = vgsvc1
pvs = /dev/loop0

[fs#app]
type = ext4
dev = /dev/mapper/vgsvc1-lvappsvc1
mnt = /svc1/app

[fs#data]
type = ext4
```

```
dev = /dev/mapper/vgsvc1-lvdatasvc1
mnt = /svc1/data
```

Le Répertoire des Scripts de Démarrage des Services

Les services sont utilisés pour gérer des applications. Si, par exemple, on souhaitait construire un service de type LAMP, nous aurons besoins de 2 scripts, un pour la base de données MySQL et l'autre pour le serveur Apache. Ces scripts doivent se trouver dans le répertoire **/etc/opensvc/svc1.dir**. Cependant, OpenSVC cherche dans le répertoire **/etc/opensvc/svc1.d** pour des scripts :

```
[root@node1 ~]# mkdir /etc/opensvc/svc1.dir
[root@node1 ~]# ln -s /etc/opensvc/svc1.dir /etc/opensvc/svc1.d
```

Service Management Facility

Afin de rendre la gestion des services plus facile, OpenSVC crée un lien symbolique **svc1** dans le répertoire **/etc/opensvc** qui pointe vers la commande **/usr/bin/svcmgr** :

```
[root@node1 ~]# ls -l /etc/opensvc
total 8
-rw-r-----. 1 root root 158 Sep 12 05:20 node.conf
lrwxrwxrwx. 1 root root 15 Sep 12 05:23 svc1 -> /usr/bin/svcmgr
-rw-r--r--. 1 root root 332 Sep 12 05:39 svc1.conf
lrwxrwxrwx. 1 root root 21 Sep 12 06:12 svc1.d -> /etc/opensvc/svc1.dir
drwxr-xr-x. 2 root root 6 Sep 12 06:09 svc1.dir
```

Le service peut donc être maintenant géré de deux façons :

```
[root@node1 ~]# svcmgr -s svc1 print status
svc1                down          warn
`- instances
  |- node2.opensvc.loc      undef      daemon down
```

```

`- node1.opensvc.loc          down      warn,
    |                          frozen,
    |                          idle
    |- ip#0                    ..... down  svc1.opensvc.loc@bond0
    |- disk#0                  ..... down  vg vgsvc1
    |- fs#app                  ..... down  ext4 /dev/mapper/vgsvc1-lvapps
    |                          vcl@/svc1/app
    |- fs#data                 ..... down  ext4 /dev/mapper/vgsvc1-lvdata
    |                          svc1@/svc1/data
    `-- sync#i0                ...0./. warn  rsync svc config to nodes
                                           warn: passive node needs
                                           update

```

ou :

```

[root@node1 ~]# /etc/opensvc/svc1 print status
svc1                      down      warn
`- instances
    |- node2.opensvc.loc   undef    daemon down
    `-- node1.opensvc.loc   down      warn,
        |                  frozen,
        |                  idle
        |- ip#0            ..... down  svc1.opensvc.loc@bond0
        |- disk#0          ..... down  vg vgsvc1
        |- fs#app          ..... down  ext4 /dev/mapper/vgsvc1-lvapps
        |                  vcl@/svc1/app
        |- fs#data         ..... down  ext4 /dev/mapper/vgsvc1-lvdata
        |                  svc1@/svc1/data
        `-- sync#i0        ...0./. warn  rsync svc config to nodes
                                           warn: passive node needs
                                           update

```

Dans la sortie de la commande ci-dessus, le statut global est **warn** parce que toutes les ressources sont **down**.

Démarrez le service :

```
[root@node1 ~]# svcl start --local
node1.opensvc.loc.svc1.ip#0      checking 192.168.121.42 availability
node1.opensvc.loc.svc1.ip#0      ifconfig bond0:1 192.168.121.42 netmask 255.255.255.0 up
node1.opensvc.loc.svc1.ip#0      send gratuitous arp to announce 192.168.121.42 is at bond0
node1.opensvc.loc.svc1.disk#0     vgchange --addtag @node1.opensvc.loc vgsvc1
node1.opensvc.loc.svc1.disk#0     output:
node1.opensvc.loc.svc1.disk#0      Volume group "vgsvc1" successfully changed
node1.opensvc.loc.svc1.disk#0
node1.opensvc.loc.svc1.disk#0     vg vgsvc1 is already up
node1.opensvc.loc.svc1.fs#app      ext4 /dev/mapper/vgsvc1-lvappsvc1@/svc1/app is already mounted
node1.opensvc.loc.svc1.fs#data     ext4 /dev/mapper/vgsvc1-lvdatasvc1@/svc1/data is already mounted
```

Cette séquence :

- vérifie si l'adresse 192.168.121.42 n'est pas déjà utilisée,
- active l'adresse IP,
- active le VG si ceci n'est pas déjà fait,
- monte chaque système de fichiers.

Arrêtez maintenant le service :

```
[root@node1 ~]# svcl stop
node1.opensvc.loc.svc1           stop action requested
```

Deémarrez le service :

```
[root@node1 ~]# svcl start --local
node1.opensvc.loc.svc1.ip#0      checking 192.168.121.42 availability
node1.opensvc.loc.svc1.ip#0      ifconfig bond0:1 192.168.121.42 netmask 255.255.255.0 up
node1.opensvc.loc.svc1.ip#0      send gratuitous arp to announce 192.168.121.42 is at bond0
node1.opensvc.loc.svc1.disk#0     vgchange --addtag @node1.opensvc.loc vgsvc1
node1.opensvc.loc.svc1.disk#0     output:
```

```
node1.opensvc.loc.svc1.disk#0      Volume group "vgsvc1" successfully changed
node1.opensvc.loc.svc1.disk#0
node1.opensvc.loc.svc1.disk#0      vgchange -a y vgsvc1
node1.opensvc.loc.svc1.disk#0      output:
node1.opensvc.loc.svc1.disk#0      2 logical volume(s) in volume group "vgsvc1" now active
node1.opensvc.loc.svc1.disk#0
node1.opensvc.loc.svc1.fs#app       e2fsck -p /dev/mapper/vgsvc1-lvappsvc1
node1.opensvc.loc.svc1.fs#app       output:
node1.opensvc.loc.svc1.fs#app       /dev/mapper/vgsvc1-lvappsvc1: clean, 12/24576 files, 8757/98304 blocks
node1.opensvc.loc.svc1.fs#app
node1.opensvc.loc.svc1.fs#app       mount -t ext4 /dev/mapper/vgsvc1-lvappsvc1 /svc1/app
node1.opensvc.loc.svc1.fs#data      e2fsck -p /dev/mapper/vgsvc1-lvdatasvc1
node1.opensvc.loc.svc1.fs#data      output:
node1.opensvc.loc.svc1.fs#data      /dev/mapper/vgsvc1-lvdatasvc1: clean, 12/24576 files, 8757/98304 blocks
node1.opensvc.loc.svc1.fs#data
node1.opensvc.loc.svc1.fs#data      mount -t ext4 /dev/mapper/vgsvc1-lvdatasvc1 /svc1/data
```

Notez que le processus de démarrage a monté les deux LV :

```
[root@node1 ~]# mount | grep svc1
/dev/mapper/vgsvc1-lvappsvc1 on /svc1/app type ext4 (rw,relatime,seclabel,data=ordered)
/dev/mapper/vgsvc1-lvdatasvc1 on /svc1/data type ext4 (rw,relatime,seclabel,data=ordered)
```

et activé l'adresse IP :

```
[root@node1 ~]# ip addr list bond0
5: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP qlen 1000
    link/ether 08:00:27:59:5a:01 brd ff:ff:ff:ff:ff:ff
    inet 192.168.121.1/24 brd 192.168.121.255 scope global bond0
        valid_lft forever preferred_lft forever
    inet 192.168.121.42/24 brd 192.168.121.255 scope global secondary bond0:1
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe59:5a01/64 scope link tentative dadfailed
```

```
valid_lft forever preferred_lft forever
```

Notre service fonctionne maintenant sur node1 :

```
[root@node1 ~]# svcmgr -s svc1 print status
svc1                                up
`- instances
  |- node2.opensvc.loc             undef    daemon down
  `-- node1.opensvc.loc            up       frozen,
      |                             idle,
      |                             started
      |- ip#0                      ..... up   svc1.opensvc.loc@bond0
      |- disk#0                   ..... up   vg vgsvc1
      |- fs#app                   ..... up   ext4 /dev/mapper/vgsvc1-lvapps
      |                             vcl@/svc1/app
      |- fs#data                  ..... up   ext4 /dev/mapper/vgsvc1-lvdata
      |                             svc1@/svc1/data
      `-- sync#i0                 ...0./. n/a   rsync svc config to nodes
                                          info: paused, service not up
```

Intégrer une Application

Applications launcher directory

Déplacez maintenant le répertoire de lancement des applications sur un LV géré par le cluster :

```
[root@node1 ~]# cd /etc/opensvc/
[root@node1 opensvc]# ls -lart | grep svc1
lrwxrwxrwx.  1 root root  15 Sep 12 05:23 svc1 -> /usr/bin/svcmgr
drwxr-xr-x.  2 root root   6 Sep 12 06:09 svc1.dir
lrwxrwxrwx.  1 root root  21 Sep 12 06:12 svc1.d -> /etc/opensvc/svc1.dir
-rw-r--r--.  1 root root 358 Sep 12 07:44 svc1.conf
```

```
[root@node1 opensvc]# rm -f svc1.d
[root@node1 opensvc]# rmdir svc1.dir
[root@node1 opensvc]# mkdir /svc1/app/init.d
[root@node1 opensvc]# ln -s /svc1/app/init.d svc1.d
[root@node1 opensvc]# ls -lart | grep svc1
lrwxrwxrwx.    1 root root   15 Sep 12 05:23 svc1 -> /usr/bin/svcmgr
-rw-r--r--.    1 root root  358 Sep 12 07:44 svc1.conf
lrwxrwxrwx.    1 root root   16 Sep 12 08:44 svc1.d -> /svc1/app/init.d
```

Obtenir le binaire de l'application

Téléchargez l'application **mongoose** - un serveur web léger :

```
[root@node1 opensvc]# cd /svc1/app
[root@node1 app]# wget -O /svc1/app/webserver
https://storage.googleapis.com/google-code-archive-downloads/v2/code.google.com/mongoose/mongoose-lua-sqlite-ssl-
static-x86_64-5.1
--2018-09-12 08:45:53--
https://storage.googleapis.com/google-code-archive-downloads/v2/code.google.com/mongoose/mongoose-lua-sqlite-ssl-
static-x86_64-5.1
Resolving storage.googleapis.com (storage.googleapis.com)... 172.217.22.144, 2a00:1450:4007:813::2010
Connecting to storage.googleapis.com (storage.googleapis.com)|172.217.22.144|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2527016 (2.4M) [application/octet-stream]
Saving to: '/svc1/app/webserver'

100%[=====>] 2,527,016   2.75MB/s   in 0.9s

2018-09-12 08:45:55 (2.75 MB/s) - '/svc1/app/webserver' saved [2527016/2527016]

[root@node1 app]# ls -l /svc1/app/webserver
-rw-r--r--. 1 root root 2527016 Jan 23  2016 /svc1/app/webserver
```

Rendez exécutable le fichier **/svc1/app/webserver** :

```
[root@node1 app]# chmod u+x /svc1/app/webserver
```

Créez le fichier index.html :

```
[root@node1 app]# cd /svc1/data/  
[root@node1 data]# vi index.html  
[root@node1 data]# cat index.html  
<html>  
<body>It Works !</body>  
</html>
```

Créez maintenant un script de gestion de l'application :

```
[root@node1 data]# cd /svc1/app/init.d  
[root@node1 init.d]# vi weblauncher  
[root@node1 init.d]# cat weblauncher  
#!/bin/bash  
  
SVCR00T=/svc1  
APPR00T=${SVCR00T}/app  
DAEMON=${APPR00T}/webserver  
DAEMON_BASE=$(basename $DAEMON)  
DAEMONOPTS="-document_root ${SVCR00T}/data -index_files index.html -listening_port 8080"  
  
function status {  
    pgrep $DAEMON_BASE >/dev/null 2>&1  
}  
  
case $1 in  
restart)  
    killall $DAEMON_BASE  
    $DAEMON
```

```
start)    ;;
status && {
    echo "already started"
    exit 0
}
nohup $DAEMON $DAEMONOPTS >> /dev/null 2>&1 &
;;
stop)
killall $DAEMON_BASE
;;
info)
echo "Name: webserver"
;;
status)
status
exit $?
;;
*)
echo "unsupported action: $1" >&2
exit 1
;;
esac
```

Rendez ce script exécutable :

```
[root@node1 init.d]# chmod u+x weblauncher
```

Testez le script :

```
[root@node1 init.d]# ./weblauncher status
[root@node1 init.d]# echo $?
1
[root@node1 init.d]# ./weblauncher start
```

```
[root@node1 init.d]# ./weblauncher status
[root@node1 init.d]# echo $?
0
[root@node1 init.d]# ./weblauncher stop
[root@node1 init.d]# ./weblauncher status
[root@node1 init.d]# echo $?
1
```

Editez maintenant le fichier **/etc/opensvc/svc1.conf** :

```
[root@node1 ~]# svc1 edit config
[root@node1 ~]# cat /etc/opensvc/svc1.conf
[DEFAULT]
id = 7adc8d25-1668-4078-a3f0-5b4472c436d2
app = MyApp
nodes = node1.opensvc.loc node2.opensvc.loc

[ip#0]
ipname = svc1.opensvc.loc
ipdev = bond0

[disk#0]
type = vg
name = vgsvc1
pvs = /dev/loop0

[fs#app]
type = ext4
dev = /dev/mapper/vgsvc1-lvappsvc1
mnt = /svc1/app

[fs#data]
type = ext4
dev = /dev/mapper/vgsvc1-lvdatasvc1
```

```
mnt = /svc1/data

[app#web]
script = weblauncher
start = 10
check = 10
stop = 90
```

Cette configuration indique à OpenSVC d'appeler le script weblauncher avec l'argument :

- **start** quand le service OpenSVC démarre,
- **stop** quand le service OpenSVC s'arrête,
- **status** quand le service OpenSVC a besoin de connaître le statut de l'application.

Démarrez maintenant le service **svc1** :

```
[root@node1 ~]# svc1 start --local
node1.opensvc.loc.svc1.ip#0      192.168.121.42 is already up on bond0
node1.opensvc.loc.svc1.disk#0    vg vgsvc1 is already up
node1.opensvc.loc.svc1.fs#app    ext4 /dev/mapper/vgsvc1-lvappsvc1@/svc1/app is already mounted
node1.opensvc.loc.svc1.fs#data   ext4 /dev/mapper/vgsvc1-lvdatasvc1@/svc1/data is already mounted
node1.opensvc.loc.svc1.app#web   exec /svc1/app/init.d/weblauncher start as user root
node1.opensvc.loc.svc1.app#web   start done in 0:00:01.011693 ret 0
```

En regardant le statut du service, on note que l'application a démarré :

```
[root@node1 ~]# svc1 print status
svc1                                up
`- instances
  |- node2.opensvc.loc             undef    daemon down
  `- node1.opensvc.loc             up       frozen, idle, started
    |- ip#0                        ..... up   svc1.opensvc.loc@bond0
    |- disk#0                      ..... up   vg vgsvc1
    |- fs#app                      ..... up   ext4 /dev/mapper/vgsvc1-lvappsvc1@/svc1/app
```

```

|- fs#data          ..... up          ext4 /dev/mapper/vgsvc1-lvdatasvc1@/svc1/data
|- app#web           ...../. up        forking: weblauncher
`- sync#i0          ...0./.. up        rsync svc config to nodes

```

Pour vérifier si c'est vraiment le cas, utilisez les commandes suivantes :

```

[root@node1 ~]# ps auxww | grep web
root      22374  0.0  0.0   2540   328 pts/1    S    09:56   0:00 /svc1/app/webserver -document_root /svc1/data -
index_files index.html -listening_port 8080
root      23406  0.0  0.1 112660   972 pts/1    R+   09:58   0:00 grep --color=auto web
[root@node1 ~]# wget -q0 - http://svc1.opensvc.loc:8080/
<html>
<body>It Works !</body>
</html>

```

Arrêtez maintenant le service svc1 :

```

[root@node1 ~]# svc1 stop --local
node1.opensvc.loc.svc1.app#web      exec /svc1/app/init.d/weblauncher stop as user root
node1.opensvc.loc.svc1.app#web      stop done in 0:00:01.016128 ret 0
node1.opensvc.loc.svc1.fs#data      umount /svc1/data
node1.opensvc.loc.svc1.fs#app       umount /svc1/app
node1.opensvc.loc.svc1.disk#0       vgchange --deltag @node1.opensvc.loc vgsvc1
node1.opensvc.loc.svc1.disk#0       output:
node1.opensvc.loc.svc1.disk#0       Volume group "vgsvc1" successfully changed
node1.opensvc.loc.svc1.disk#0
node1.opensvc.loc.svc1.disk#0       vgchange -a n vgsvc1
node1.opensvc.loc.svc1.disk#0       output:
node1.opensvc.loc.svc1.disk#0       0 logical volume(s) in volume group "vgsvc1" now active
node1.opensvc.loc.svc1.disk#0
node1.opensvc.loc.svc1.ip#0         ifconfig bond0:1 down
node1.opensvc.loc.svc1.ip#0         checking 192.168.121.42 availability

```

Cette commande :

- arrête l'application,
- démonte les systèmes de fichiers,
- désactive le VG,
- désactive l'adresse IP.

Pour vérifier le statut, utilisez la commande suivante :

```
[root@node1 ~]# svcl print status
svcl                                down
`- instances
  |- node2.opensvc.loc              undef    daemon down
  `- node1.opensvc.loc              down     frozen, idle
    |- ip#0                          ..... down    svcl.opensvc.loc@bond0
    |- disk#0                        ..... down    vg vgsvc1
    |- fs#app                         ..... down    ext4 /dev/mapper/vgsvc1-lvappsvc1@/svcl/app
    |- fs#data                       ..... down    ext4 /dev/mapper/vgsvc1-lvdatasvc1@/svcl/data
    |- app#web                       ...../. down    forking: weblauncher
    `- sync#i0                       ...0./. up      rsync svc config to nodes
```

Démarrez de nouveau le service :

```
[root@node1 ~]# svcl start --local
node1.opensvc.loc.svc1.ip#0        checking 192.168.121.42 availability
node1.opensvc.loc.svc1.ip#0        ifconfig bond0:1 192.168.121.42 netmask 255.255.255.0 up
node1.opensvc.loc.svc1.ip#0        send gratuitous arp to announce 192.168.121.42 is at bond0
node1.opensvc.loc.svc1.disk#0       vgchange --addtag @node1.opensvc.loc vgsvc1
node1.opensvc.loc.svc1.disk#0       output:
node1.opensvc.loc.svc1.disk#0       Volume group "vgsvc1" successfully changed
node1.opensvc.loc.svc1.disk#0       vgchange -a y vgsvc1
node1.opensvc.loc.svc1.disk#0       output:
node1.opensvc.loc.svc1.disk#0       2 logical volume(s) in volume group "vgsvc1" now active
node1.opensvc.loc.svc1.disk#0
node1.opensvc.loc.svc1.fs#app       e2fsck -p /dev/mapper/vgsvc1-lvappsvc1
```

```
node1.opensvc.loc.svc1.fs#app      output:
node1.opensvc.loc.svc1.fs#app      /dev/mapper/vgsvc1-lvappsvc1: clean, 15/24576 files, 11228/98304 blocks
node1.opensvc.loc.svc1.fs#app
node1.opensvc.loc.svc1.fs#app      mount -t ext4 /dev/mapper/vgsvc1-lvappsvc1 /svc1/app
node1.opensvc.loc.svc1.fs#data      e2fsck -p /dev/mapper/vgsvc1-lvdatasvc1
node1.opensvc.loc.svc1.fs#data      output:
node1.opensvc.loc.svc1.fs#data      /dev/mapper/vgsvc1-lvdatasvc1: clean, 13/24576 files, 8758/98304 blocks
node1.opensvc.loc.svc1.fs#data
node1.opensvc.loc.svc1.fs#data      mount -t ext4 /dev/mapper/vgsvc1-lvdatasvc1 /svc1/data
node1.opensvc.loc.svc1.app#web      exec /svc1/app/init.d/weblauncher start as user root
node1.opensvc.loc.svc1.app#web      start done in 0:00:01.009806 ret 0
```

Service Failover

Le service s'exécute sur node1.

Ajoutez maintenant node2 au cluster de node1 :

```
[root@node1 ~]# nodemgr get --param cluster.secret
6e828240b63811e8819b080027415bd7

[root@node2 ~]# nodemgr daemon join --secret 6e828240b63811e8819b080027415bd7 --node node1.opensvc.loc
node2.opensvc.loc  freeze local node
node2.opensvc.loc  update node.env => TST
node2.opensvc.loc  update heartbeat hb#1
node2.opensvc.loc  join node node1.opensvc.loc
node2.opensvc.loc  thaw local node
```

Notez maintenant que les deux noeuds se trouvent dans le cluster :

```
[root@node2 ~]# svcmon

Threads                node1.opensvc.loc node2.opensvc.loc
```

```

hb#1.rx    running 0.0.0.0:10000 | 0      /
hb#1.tx    running                | 0      /
listener   running 0.0.0.0:1214
monitor    running
scheduler  running

Nodes
score      | 91      92
load 15m   | 0.1     0.1
mem        | 56/98%:488m  53/98%:488m
swap       | 1/90%:1g    1/90%:1g
state      |

Services
svc1       up!      failover | 0*      !!P
node1.opensvc.loc node2.opensvc.loc

```

Synchronisez les deux noeuds :

```

[root@node1 ~]# svc1 sync nodes
node1.opensvc.loc.svc1.sync#i0    transfered 279 B at 598 B/s
node1.opensvc.loc.svc1            request action 'postsync --waitlock=3600' on node node2.opensvc.loc

```

La réplication n'a lieu que dans le où les conditions suivantes soient remplies :

- le nouveau noeud est déclaré dans le fichier **/etc/opensvc/svc1.conf**,
- le noeud récepteur a confiance dans le noeud émetteur (via ssh),
- le service svc1 soit activé et fonctionnel sur le node1,
- la synchronisation précédente est plus ancienne.

Consultez maintenant le contenu du répertoire **/etc/opensvc** sur node2 :

```

[root@node2 ~]# ls -l /etc/opensvc
total 8
-rw----- . 1 root root 195 Sep 12 10:21 node.conf

```

```
lrwxrwxrwx. 1 root root 15 Sep 12 05:23 svc1 -> /usr/bin/svcmgr
-rw-r--r--. 1 root root 421 Sep 12 09:52 svc1.conf
lrwxrwxrwx. 1 root root 16 Sep 12 08:44 svc1.d -> /svc1/app/init.d
drwxr-xr-x. 2 root root 6 Sep 12 06:09 svc1.dir
```

Notez que lien symbolique orphelin **svc1.d** clignote.

En comparant ce contenu avec celui du node1, on constate l'absence du répertoire **svc1.dir** :

```
[root@node1 ~]# ls -l /etc/opensvc
total 8
-rw-r-----. 1 root root 176 Sep 12 10:11 node.conf
lrwxrwxrwx. 1 root root 15 Sep 12 05:23 svc1 -> /usr/bin/svcmgr
-rw-r--r--. 1 root root 421 Sep 12 09:52 svc1.conf
lrwxrwxrwx. 1 root root 16 Sep 12 08:44 svc1.d -> /svc1/app/init.d
```

Supprimez donc le répertoire **svc1.dir** sur node2 et créez le répertoire **/svc1/app/init.d** :

```
[root@node2 opensvc]# rmdir svc1.dir
[root@node2 opensvc]# mkdir /svc1/app/init.d
[root@node2 opensvc]# ls -lart | grep svc1
lrwxrwxrwx. 1 root root 15 Sep 12 05:23 svc1 -> /usr/bin/svcmgr
lrwxrwxrwx. 1 root root 16 Sep 12 08:44 svc1.d -> /svc1/app/init.d
-rw-r--r--. 1 root root 421 Sep 12 09:52 svc1.conf
```

Installez maintenant l'application **mongoose** sur node2 :

```
[root@node2 opensvc]# cd /svc1/app
[root@node2 app]# wget -O /svc1/app/webserver
https://storage.googleapis.com/google-code-archive-downloads/v2/code.google.com/mongoose/mongoose-lua-sqlite-ssl-static-x86_64-5.1
--2018-09-12 11:36:24--
https://storage.googleapis.com/google-code-archive-downloads/v2/code.google.com/mongoose/mongoose-lua-sqlite-ssl-static-x86_64-5.1
```

```
Resolving storage.googleapis.com (storage.googleapis.com)... 216.58.204.240, 2a00:1450:4007:80c::2010
Connecting to storage.googleapis.com (storage.googleapis.com)|216.58.204.240|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2527016 (2.4M) [application/octet-stream]
Saving to: '/svc1/app/webserver'
```

```
100%[=====>] 2,527,016 2.45MB/s in 1.0s
```

```
2018-09-12 11:36:25 (2.45 MB/s) - '/svc1/app/webserver' saved [2527016/2527016]
```

```
[root@node2 app]# ls -l /svc1/app/webserver
-rw-r--r--. 1 root root 2527016 Jan 23 2016 /svc1/app/webserver
```

Rendez **/svc1/app/webserver** exécutable :

```
[root@node2 app]# chmod u+x /svc1/app/webserver
```

Créez le fichier index.html :

```
[root@node2 app]# cd /svc1/data/
[root@node2 data]# vi index.html
[root@node2 data]# cat index.html
<html>
<body>It Works !</body>
</html>
```

ainsi que le fichier **/svc1/app/init.d/weblauncher** :

```
[root@node2 data]# cd /svc1/app/init.d
[root@node2 init.d]# vi weblauncher
[root@node2 init.d]# cat weblauncher
#!/bin/bash
```

```
SVCR00T=/svc1
```

```
APPROOT=${SVCROOT}/app
DAEMON=${APPROOT}/webserver
DAEMON_BASE=$(basename $DAEMON)
DAEMONOPTS="-document_root ${SVCROOT}/data -index_files index.html -listening_port 8080"

function status {
    pgrep $DAEMON_BASE >/dev/null 2>&1
}

case $1 in
restart)
    killall $DAEMON_BASE
    $DAEMON
    ;;
start)
    status && {
        echo "already started"
        exit 0
    }
    nohup $DAEMON $DAEMONOPTS >> /dev/null 2>&1 &
    ;;
stop)
    killall $DAEMON_BASE
    ;;
info)
    echo "Name: webserver"
    ;;
status)
    status
    exit $?
    ;;
*)
    echo "unsupported action: $1" >&2
    exit 1

```

```
;;  
esac
```

Rendez le script exécutable :

```
[root@node2 init.d]# chmod u+x weblauncher
```

Testez le script :

```
[root@node2 init.d]# ./weblauncher status  
[root@node2 init.d]# echo $?  
1  
[root@node2 init.d]# ./weblauncher start  
[root@node2 init.d]# ./weblauncher status  
[root@node2 init.d]# echo $?  
0  
[root@node2 init.d]# ./weblauncher stop  
[root@node2 init.d]# ./weblauncher status  
[root@node2 init.d]# echo $?  
1
```

Arrêtez maintenant le service svc1 sur node1 :

```
[root@node1 ~]# svcmgr -s svc1 stop  
node1.opensvc.loc.svc1          stop action requested
```

Démarrez le service sur node2 :

```
[root@node2 ~]# svc1 start --local  
node2.opensvc.loc.svc1.ip#0      checking 192.168.121.42 availability  
node2.opensvc.loc.svc1.ip#0      ifconfig bond0:1 192.168.121.42 netmask 255.255.255.0 up  
node2.opensvc.loc.svc1.ip#0      send gratuitous arp to announce 192.168.121.42 is at bond0  
node2.opensvc.loc.svc1.disk#0     vg vgsvc1 is already up  
node2.opensvc.loc.svc1.fs#app     ext4 /dev/mapper/vgsvc1-lvappsvc1@/svc1/app is already mounted
```

```
node2.opensvc.loc.svc1.fs#data      ext4 /dev/mapper/vgsvc1-lvdatasvc1@svc1/data is already mounted
node2.opensvc.loc.svc1.app#web      exec /svc1/app/init.d/weblauncher start as user root
node2.opensvc.loc.svc1.app#web      start done in 0:00:01.008940 ret 0
```

Haute Disponibilité

Afin de mettre en place la haute disponibilité, il convient de changer le mode d'orchestration à **ha**. Editez donc le fichier **/etc/opensvc/svc1.conf** sur node2 en ajoutant **orchestrate = ha** dans la section **[DEFAULT]** et **monitor = True** dans la section **[app#web]** :

```
[root@node2 ~]# svc1 edit config
[root@node2 ~]# cat /etc/opensvc/svc1.conf
[DEFAULT]
id = 7adc8d25-1668-4078-a3f0-5b4472c436d2
app = MyApp
nodes = node1.opensvc.loc node2.opensvc.loc
orchestrate = ha

[ip#0]
ipname = svc1.opensvc.loc
ipdev = bond0

[disk#0]
type = vg
name = vgsvc1
pvs = /dev/loop0

[fs#app]
type = ext4
dev = /dev/mapper/vgsvc1-lvappsvc1
mnt = /svc1/app

[fs#data]
```

```
type = ext4
dev = /dev/mapper/vgsvc1-lvdatasvc1
mnt = /svc1/data
```

```
[app#web]
monitor = True
script = weblauncher
start = 10
check = 10
stop = 90
```

Siasissez la commande suivante puis testez la HA :

```
[root@node2 ~]# svc1 thaw
```

<html> <DIV ALIGN="CENTER"> Copyright © 2020 Hugh Norris. </DIV> </html>
