

Compte rendu d'Installation « Keepalived »

Table des matières

Objectif(s) :	2
Légende :	2
Keepalived :	3
1) Configuration :	3
2) Utilisation de keepalived	6

Objectif(s) :

L'objectif de ce tuto est de configurer Keepalived afin d'avoir une haute disponibilité en équilibrage de charge.

Légende :

- Les commandes ou les chemins (absolue/relatif) sont en gras, souligné et en italique ex :

➤ ***Apt-get update***

- Des captures d'écrans ont été prises afin de faciliter la compréhension du lecteur.

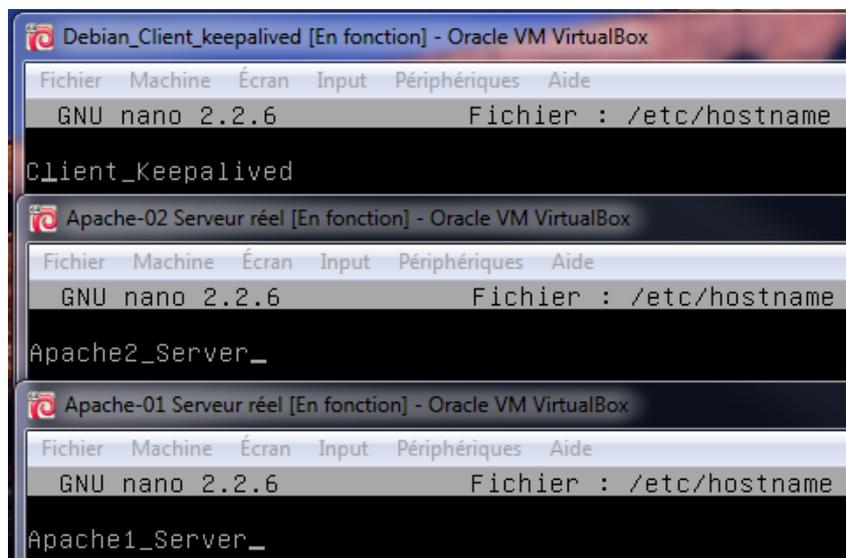
Machine	Os	Distribution	Version	C/S	IP
POSTE21	Debian	Linux	8.5	S	192.168.1.140 Vip : 192.168.1.143
POSTE21	Debian	Linux	8.5	S	192.168.1.141

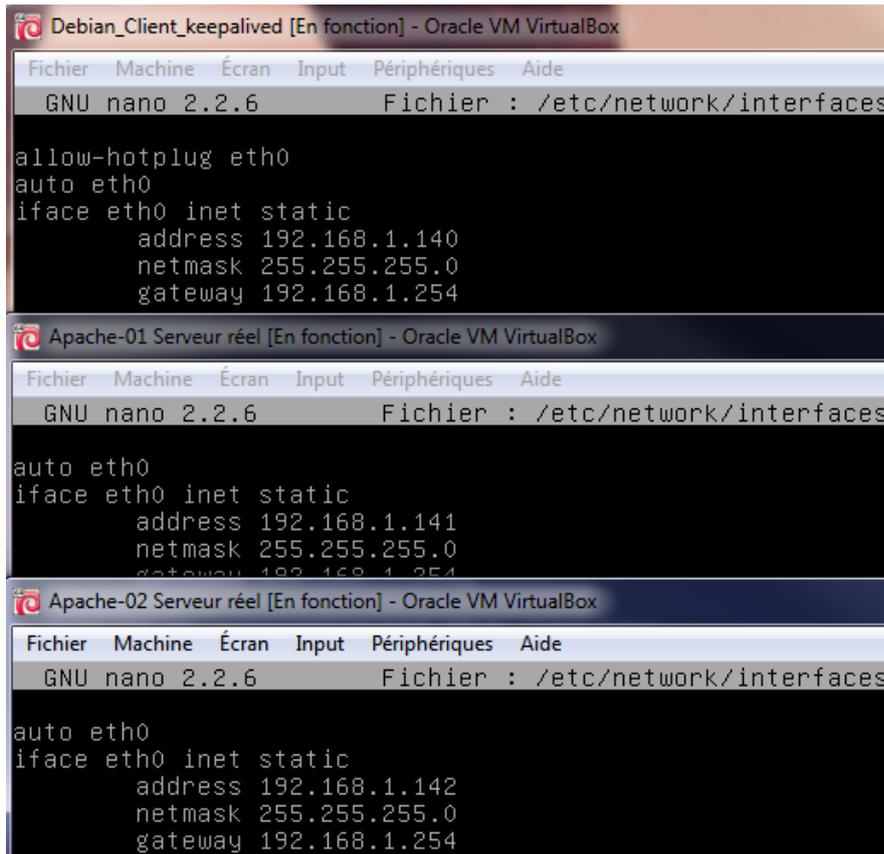
POSTE21	Debian	Linux	8.5	S	192.168.1.142
---------	--------	-------	-----	---	---------------

Keepalived :

1) Configuration :

On va configurer pour commencer les 3 machines :





```
Debian_Client_keepalived [En fonction] - Oracle VM VirtualBox
Fichier  Machine  Écran  Input  Périphériques  Aide
GNU nano 2.2.6      Fichier : /etc/network/interfaces

allow-hotplug eth0
auto eth0
iface eth0 inet static
    address 192.168.1.140
    netmask 255.255.255.0
    gateway 192.168.1.254

Apache-01 Serveur réel [En fonction] - Oracle VM VirtualBox
Fichier  Machine  Écran  Input  Périphériques  Aide
GNU nano 2.2.6      Fichier : /etc/network/interfaces

auto eth0
iface eth0 inet static
    address 192.168.1.141
    netmask 255.255.255.0
    gateway 192.168.1.254

Apache-02 Serveur réel [En fonction] - Oracle VM VirtualBox
Fichier  Machine  Écran  Input  Périphériques  Aide
GNU nano 2.2.6      Fichier : /etc/network/interfaces

auto eth0
iface eth0 inet static
    address 192.168.1.142
    netmask 255.255.255.0
    gateway 192.168.1.254
```

*Commençons l'installation, Nous allons installer ipvsadm
Apt-get install ipvsadm*

Sur ce même serveur on s'appuie sur une adresse IP Virtuelle qu'il faut déclarer au niveau de l'interface réseau du directeur.

```
auto bond0
iface bond0 inet static
    address 192.168.1.140
    netmask 255.255.255.255
    broadcast 192.168.1.255
    network 192.168.1.0
```

Pour configurer l'interface il faut rajouter cette ligne dans */etc/modprobe.d/bon.conf*



```
192.168.1.140 - PuTTY
GNU nano 2.2.6      Fichier : bond.conf

alias bond0 bonding
```

On fait ensuite un *ipup bond0* pour l'activer:

```
root@debianJeudi:/etc/modprobe.d# ipup bond0
root@debianJeudi:/etc/modprobe.d#
```

L'adresse IP Virtuelle est partagée entre le directeur et les serveurs physiques. Mais il n'y a que le directeur qui répond aux requêtes adressées sur l'adresse VIP. Cela explique pourquoi, cette dernière est de type /32, ne laissant qu'un seul hôte disponible sur le sous-réseau. La déclaration de ce genre d'adresse s'effectue grâce à la commande ipvsadm :

```
Ipvsadm -A -t 192.168.1.143:80 -s rr
```

Ou -A permet d'ajouter un service

Ou -t précise le protocole TCP suivi de l'adresse IP du service et le port

Ou -s précise la politique de répartition, ici c'est le rr (Round Robin)

```
root@debianJeudi:/etc/modprobe.d# ipvsadm -A -t 192.168.1.143:80 -s rr
```

Il faut maintenant ajouter les serveurs réels proposant le service web :

```
root@debianJeudi:/etc/modprobe.d# ipvsadm -a -t 192.168.1.143:80 -r 192.168.1.141:80 -g -w
root@debianJeudi:/etc/modprobe.d# ipvsadm -a -t 192.168.1.143:80 -r 192.168.1.142:80 -g -w
root@debianJeudi:/etc/modprobe.d# ipvsadm -ln
IP Virtual Server version 1.2.1 (size=4096)
Prot LocalAddress:Port Scheduler Flags
  -> RemoteAddress:Port      Forward Weight ActiveConn InActConn
TCP  192.168.1.143:80 rr
  -> 192.168.1.141:80        Route    1      0          0
  -> 192.168.1.142:80        Route    1      0          0
```

Ici -a permet d'ajouter un serveur réel et l'option -t précise le protocole TCP suivi de son adresse IP ainsi que du port d'écoute. L'option -r spécifie l'adresse du serveur réel et son port d'écoute. L'option -g signifie que l'on est en mode passerelle (gateway aussi appelé Direct Routing). Et enfin l'option -w permet d'indiquer le poids.

Maintenant que le directeur est prêt à fonctionner, il ne reste plus qu'à configurer les serveurs réels pour y déclarer les adresses virtuelle VIP.

On va d'abord s'assurer que les serveurs réels ne répondent pas aux requêtes ARP qui leur sont adressées, car seules les requêtes venant du directeur doivent être interprétées par le VIP

On va donc éditer le fichier */etc/sysctl.conf* pour y ajouter les lignes suivantes :

```
net.ipv4.conf.all.arp_ignore=1
net.ipv4.conf.all.arp_announce=2
net.ipv4.conf.lo.arp_ignore=1
net.ipv4.conf.lo.arp_announce=2
```

On va ensuite recharger la configuration du noyau en exécutant la commande suivante **dans les serveurs webs**. ??? serveurs web ou maître ???

```
root@debianJeudi:/etc/modprobe.d# sysctl -p
net.ipv4.conf.all.arp_ignore = 1
net.ipv4.conf.all.arp_announce = 2
net.ipv4.conf.lo.arp_ignore = 1
net.ipv4.conf.lo.arp_announce = 2
```

Et on va sauvegarder la configuration du directeur :

```
root@debianJeudi:/etc/modprobe.d# ipvsadm -Sn > /etc/ipvsadm_rules
```

Normalement le fichier ipvsadm_rules est systématiquement rechargé au démarrage du système d'exploitation Debian. Pour vérifier que LVS fonctionne, on peut simplement utiliser le fichier de test index.html des serveurs web en les personnalisant afin de les différencier

2) Utilisation de keepalived

apt-get install keepalived

On va éditer le fichier /etc/keepalived.conf et noter :

```
192.168.1.141 - PuTTY
GNU nano 2.2.6      Fichier : keepalived.conf

#Définition des paramètres généraux pour le serveur virtuel Linux
global_defs {
#Message de réception des notifications LVS
notification_email { admin@dmn.org
}
#en-tête "MAIL FROM!"des notifications LVS
notification_email_from admlvs@dmn.org
#serveur SMP pour les notifications
smtp_server smtp.sio.fr
#timeout pour le SMTP
smtp_connect_timeout 30
#Nom du LVS
router_id LVS_WEB
}
#Identifiant pour le groupe VRRP
vrrp_sync_group KAD-VRRP {
#Définition des instances VRRP du groupe
group{ KEEP1
}
#script de notification pour les actions du groupe VRRP
notify /root/adm/notify_ka.sh
}
#Définition d'une instance VRRP
vrrp_instance KEEP1{
#état BACKUP sur tous pour éviter les variations
state BACKUP
#interface à vérifier
interface eth0
#id du VRRP
virtual_router_id 50
#authentification
authentication {
auth_type uadmin
auth_pass root
}
#pas de passe-droit (pas de préemption)
nopreempt
#Priorité différence entre les fichiers de configuration
priority 100
#précise l'intervalle en secondes (set to 1)
advert_int 1
#blocage des adresses ip virtuelles
virtual_ipaddress {
192.168.1.143
}
}
#Définition du LVS
virtual_server 192.168.1.143 80 {
delay_loop 4
lb_algo wlc
lb_kind DR
persistence_timeout 120
protocol TCP

```

```
#definition du seveur apachel
real_server 192.168.1.141 80 {
#poids du serveur si besoin de préférence
weight 1
#vérification de la connexion grâce au test HTTP_GET
HTTP_GET {
url {
path /service.txt
digest
}
connect_port 80
connect_timeout 2
nb_get_retry 1
delay_before_retry 1
}
}

#definition du seveur apache2
real_server 192.168.1.142 80 {
weight1
#vérification de la connexion grâce au test HTTP_GET
HTTP_GET {
url {
path /service.txt
digest
}
connect_port80
connect_timeout 2
nb_get_retry 1
delay_before_retry 1
}
}

```

Une fois redémarré bien sûr il faut redémarrer le service :

systemctl restart keepalived

La seule différence au niveau du fichier de configuration du premier et du second serveur se trouve dans la priorité attribué au service, qu'il convient de modifier

- ➔ Priority 50
 - Dans le vrrp_instance KEEP1

```
#Définition d'une instance VRRP
vrrp_instance KEEP1{
    #état BACKUP sur tous pour éviter les variations
    state BACKUP
    #interface à vérifier
    interface eth0
    #id du VRRP
    virtual_router_id 50
    #authentification
    authentication {
        auth_type uadmin
        auth_pass root
    }

    #pas de passe-droit (pas de préemption)
    nopreempt
    #Priorité différence entre les fichiers de configuration
    priority 50
    #précise l'intervalle en secondes (set to 1)
    advert_int 1
    #blocage des adresses ip virtuelles
    virtual_ipaddress {
        192.168.1.143
    }
}
```

Grâce à la section déclarative testant les services web, on peut facilement configurer les serveurs réels et les intégrer à KeepAlived. En effet, il suffit de lui fournir les requêtes à tester de façon régulière (défini par le paramètre delay_loop et de calculer le hash (ici, il s'agit du hash d'accès à la page service.txt) grâce à la commande genhash suivante :

genhash -s 192.168.1.141 -p 80 -u /service.txt -v

```
root@Apache1_Server:/etc/keepalived# genhash -s 192.168.1.141 -p 80 -u /service.txt -v
-----[ HTTP Header Buffer ]-----
0000 48 54 54 50 2f 31 2e 31 - 20 34 30 34 20 4e 6f 74 HTTP/1.1 404 Not
0010 20 46 6f 75 6e 64 0d 0a - 44 61 74 65 3a 20 46 72 Found..Date: Fr
0020 69 2c 20 32 33 20 53 65 - 70 20 32 30 31 36 20 31 i, 23 Sep 2016 1
0030 33 3a 31 34 3a 33 36 20 - 47 4d 54 0d 0a 53 65 72 3:14:36 GMT..Ser
0040 76 65 72 3a 20 41 70 61 - 63 68 65 2f 32 2e 34 2e ver: Apache/2.4.
0050 31 30 20 28 44 65 62 69 - 61 6e 29 0d 0a 43 6f 6e 10 (Debian)..Con
0060 74 65 6e 74 2d 4c 65 6e - 67 74 68 3a 20 32 38 38 tent-Length: 288
0070 0d 0a 43 6f 6e 6e 65 63 - 74 69 6f 6e 3a 20 63 6c ..Connection: cl
0080 6f 73 65 0d 0a 43 6f 6e - 74 65 6e 74 2d 54 79 70 ose..Content-Typ
0090 65 3a 20 74 65 78 74 2f - 68 74 6d 6c 3b 20 63 68 e: text/html; ch
00a0 61 72 73 65 74 3d 69 73 - 61 6f 2d 38 38 35 39 2d 31 arset=iso-8859-1
00b0 0d 0a 0d 0a - .....
-----[ HTTP Header Ascii Buffer ]-----
HTTP/1.1 404 Not Found
Date: Fri, 23 Sep 2016 13:14:36 GMT
Server: Apache/2.4.10 (Debian)
Content-Length: 288
Connection: close
Content-Type: text/html; charset=iso-8859-1
-----[ HTML Buffer ]-----
0000 3c 21 44 4f 43 54 59 50 - 45 20 48 54 4d 4c 20 50 <!DOCTYPE HTML P
0010 55 42 4c 49 43 20 22 2d - 2f 2f 49 45 54 46 2f 2f UBLIC "-//IETF//
0020 44 54 44 20 48 54 4d 4c - 20 32 2e 30 2f 2f 45 4e DTD HTML 2.0//EN
0030 22 3e 0a 3c 68 74 6d 6c - 3e 3c 68 65 61 64 3e 0a ">.<html><head>.
0040 3c 74 69 74 6c 65 3e 34 - 30 34 20 4e 6f 74 20 46 <title>404 Not F
0050 6f 75 6e 64 3c 2f 74 69 - 74 6c 65 3e 0a 3c 2f 68 ound</title>.</h
0060 65 61 64 3a 3c 62 6f 64 - 79 3e 0a 3c 69 31 3e 4e ead><body>.<h1>N
0070 6f 74 20 46 6f 75 6e 64 - 3c 2f 68 31 3e 0a 3c 70 ot Found</h1>.<p
0080 3e 54 68 65 20 72 65 71 - 75 65 73 74 65 64 20 55 >The requested U
0090 52 4e 20 2f 73 65 72 76 - 69 63 65 2e 74 78 74 20 RL /service.txt
00a0 77 61 73 20 6e 6f 74 20 - 66 6f 75 6e 64 20 6f 6e was not found on
00b0 20 74 68 69 73 20 73 65 - 72 76 65 72 2e 3c 2f 70 this server.</p
00c0 3e 0a 3c 68 72 3e 0a 3c - 61 64 64 72 65 73 73 3e >.<hr>.<address>
00d0 41 70 61 63 68 65 2f 32 - 2e 34 2e 31 30 20 28 44 Apache/2.4.10 (D
00e0 65 62 69 61 6e 29 20 53 - 65 72 76 65 72 20 61 74 ebian) Server at
00f0 20 31 39 32 2e 31 36 38 - 2e 31 2e 31 34 31 20 50 192.168.1.141 P
0100 6f 72 74 20 38 30 3c 2f - 61 64 64 72 65 73 73 3e ort 80</address>
0110 0a 3c 2f 62 6f 64 79 3e - 3c 2f 68 74 6d 6c 3e 0a .</body></html>.
-----[ HTML hash resulting ]-----
0000 0c 38 d7 61 1b 38 b9 f9 - a8 4a a2 d3 dd ea 8b 6c .8.a.8...J....1
-----[ HTML hash final resulting ]-----
MD5SUM = 0c38d7611b38b9f9a84aa2d3dde8b6c
Global response time for [/service.txt] =6472
```

Ou MD5SUM = 0c38d7611b38b9f9a84aa2d3ddea8b6c

On va maintenant configurer le `/etc/keepalived/keepalived.conf` :

Au niveau du digest on va mettre le MD5SUM noté au-dessus dans les deux fichiers de conf des deux serveurs soit .141 / .142 :

```
#definition du seueur apache1
real_server 192.168.1.141 80 {
    #poids du serveur si besoin de préférence
    weight 1
    #vérification de la connexion grâce au test HTTP_GET
    HTTP_GET {
        url {
            path /service.txt
            digest 0c38d7611b38b9f9a84aa2d3ddea8b6c
        }
        connect_port 80
        connect_timeout 2
        nb_get_retry 1
        delay_before_retry 1
    }
}
```

Sur le serveur web 2 : 192.168.1.142 on refait la commande pour avoir le MD5SUM du deuxième serveur :

```
-----[ HTML hash resulting ]-----
0000 a9 f7 7f 87 ba ac d2 f2 - ba dd 7a e5 3b 4a 4f cb .....z.;JO.
-----[ HTML hash final resulting ]-----
MD5SUM = a9f77f87baacd2f2badd7ae53b4a4fcb
```

Au niveau du digest on va mettre le MD5SUM noté au-dessus dans les deux fichiers de conf des deux serveurs soit .141 / .142 :

Et on a donc cela dans les deux fichiers de conf :

```
#definition du seueur apache1
real_server 192.168.1.141 80 {
    #poids du serveur si besoin de préférence
    weight 1
    #vérification de la connexion grâce au test HTTP_GET
    HTTP_GET {
        url {
            path /service.txt
            digest 0c38d7611b38b9f9a84aa2d3ddea8b6c
        }
        connect_port 80
        connect_timeout 2
        nb_get_retry 1
        delay_before_retry 1
    }
}

#definition du seueur apache2
real_server 192.168.1.142 80 {
    weight1
    #vérification de la connexion grâce au test HTTP_GET
    HTTP_GET {
        url {
            path /service.txt
            digest a9f77f87baacd2f2badd7ae53b4a4fcb
        }
        connect_port80
        connect_timeout 2
        nb_get_retry 1
        delay_before_retry 1
    }
}
```

Ne pas oublier de restart le service keepalived :

```
systemctl restart keepalived.service
```

On va maintenant retourner sur le serveur maître :

Et on va clear pour être sur le IPVSADM :

```
ipvsadm -clear
```

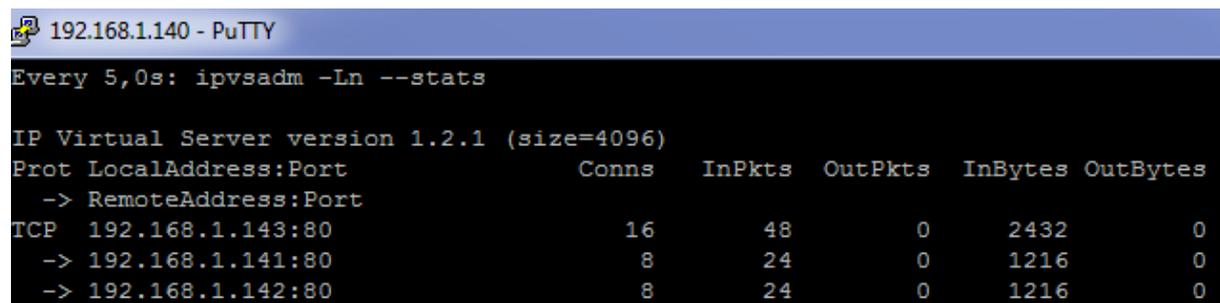
et ensuite :

```
root@Client_Keepalived:~# ipvsadm -A -t 192.168.1.143:80 -s rr
root@Client_Keepalived:~# ipvsadm -a -t 192.168.1.143:80 -r 192.168.1.141:80 -g -w 1
root@Client_Keepalived:~# ipvsadm -a -t 192.168.1.143:80 -r 192.168.1.142:80 -g -w 1
root@Client_Keepalived:~# watch -n 5 ipvsadm -Ln --stats
```

On va ensuite voir si les liens on fonctionner et si on ecoute bien les serveurs apaches sur el directeur toujours :

```
root@Client_Keepalived:~# watch -n 5 ipvsadm -Ln --stats
```

On va donc sur le LVM : 192.168.1.143 grâce au navigateur et :



The screenshot shows a PuTTY terminal window titled "192.168.1.140 - PuTTY". The terminal output displays the command "Every 5,0s: ipvsadm -Ln --stats" and the resulting statistics for IP Virtual Server version 1.2.1 (size=4096). The statistics are presented in a table format with columns for Protocol, LocalAddress:Port, RemoteAddress:Port, Conns, InPkts, OutPkts, InBytes, and OutBytes.

Prot	LocalAddress:Port	RemoteAddress:Port	Conns	InPkts	OutPkts	InBytes	OutBytes
TCP	192.168.1.143:80	->	16	48	0	2432	0
		-> 192.168.1.141:80	8	24	0	1216	0
		-> 192.168.1.142:80	8	24	0	1216	0

Il y a bien des échanges entre le maître et les serveurs apaches !