

VRRPD

Rédacteur : Frédéric Bourgeois

Le 13 novembre 2006

Màj 13/11/2008

1.1	INTRODUCTION.....	2
1.1.1	Présentation.....	2
1.1.2	Définition.....	2
2	OBJECTIF.....	2
2.1	APPLICATION.....	2
2.1.1	Redondance de serveurs grâce au protocole VRRP.....	2
2.1.2	Schéma de fonctionnement (VRRPD de base).....	3
2.2	ÉTUDE.....	4
2.2.1	Logiciels qui utilisent un principe similaire.....	4
2.2.2	Limitations des logiciels.....	4
2.3	MODIFICATIONS DE VRRPD.....	5
2.3.1	Gestion d'état globale.....	5
2.3.2	liste des modifications.....	5
3	EXPLOITATION.....	6
3.1	Compilation du binaire.....	6
3.2	Commandes VRRPD/ATROPOS.....	6
3.3	Specificités.....	8
3.3.1	Interface physique.....	8
4	VPN.....	9
4.1.1	Interface ipsec0.....	9

1.1 INTRODUCTION

1.1.1 PRÉSENTATION

Ce document présente diverses notions associées au logiciel modifié VRRPD, la version originale est disponible sur <http://vrrpd.wiki.sourceforge.net/>

1.1.2 DÉFINITION

VRRP [protocole RFC 2338] Virtual Router Redundancy Protocol. Protocole d'élection permettant de faire fonctionner plusieurs routeurs avec la même adresse IP, l'un d'eux étant le maître. Si celui-ci tombe en panne, les autres prennent le relais.

Le protocole VRRP (Virtual Router Redundancy Protocol - protocole de redondance de routeur virtuel) définit un protocole d'élection qui affecte dynamiquement la responsabilité du routage à l'un des routeurs VRRP du réseau local (routeur maître). Le processus d'élection permet un basculement dynamique de la responsabilité de routage en cas d'indisponibilité du routeur maître.

2 OBJECTIF

Mise au point d'une solution de haute disponibilité pour les pare-feux netfilter et les VPN Openswan, les sessions actives ne seront pas conservées en cas de bascule (dans l'attente de l'évolution du projet de module noyau ct_sync)

2.1 APPLICATION

2.1.1 REDONDANCE DE SERVEURS GRÂCE AU PROTOCOLE VRRP

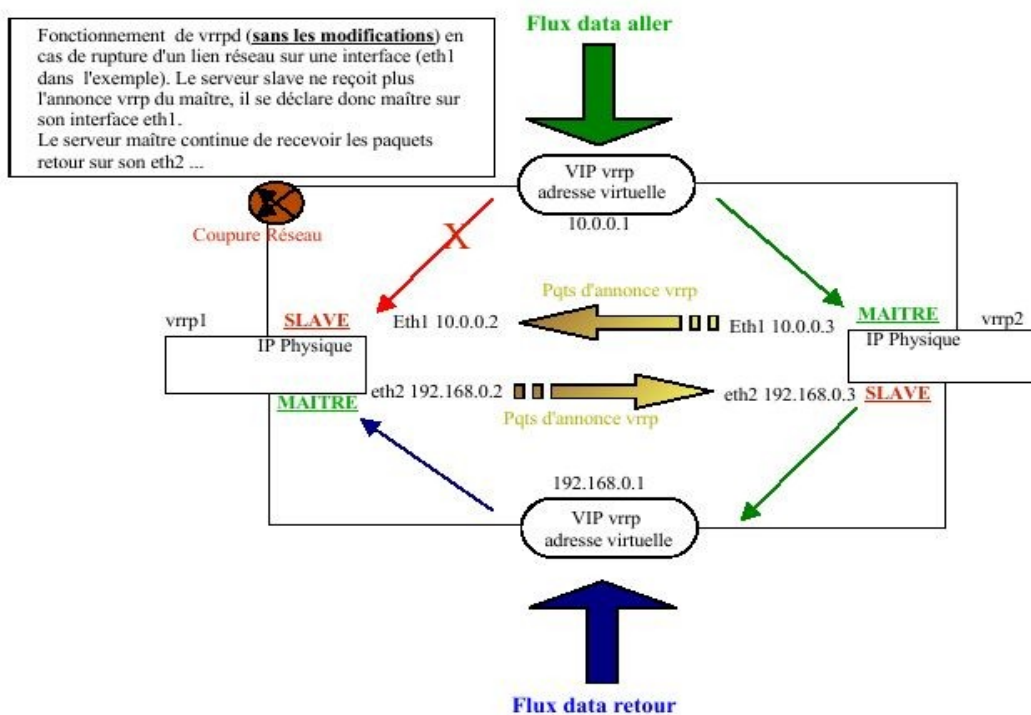
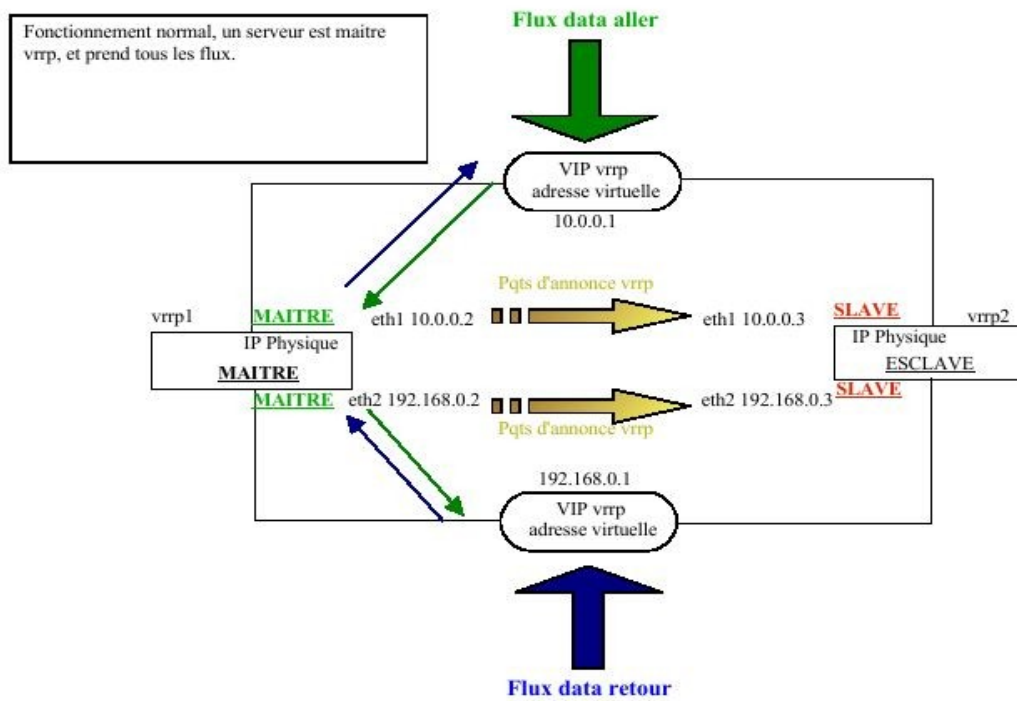
Ce document s'applique plus particulièrement à la mise en place de VRRPD avec des pare-feux Netfilter et de VPN avec Openswan, cependant l'application peut-être utilisée sur n'importe quel type de serveurs sous Linux (Apache, Bind, Postfix, etc ...)

Dans le domaine de la redondance de pare-feux, VRRPD est imparfait. Lorsqu'une machine subit une défaillance sur une interface, le démon VRRPD bascule la carte défaillante mais laisse en état les autres interfaces, les flux deviennent donc asymétriques si vous avez plusieurs cartes réseaux.

Le routage des flux de manière asymétrique ne pose pas de problème aux routeurs mais à la couche de filtrage (si elle est de type stateful) car les tables de sessions ne sont plus cohérentes.

De plus le changement d'état VRRPD ne provoque aucune action sur la machine, il n'est pas possible d'exécuter un script ou un binaire lors de la bascule.

2.1.2 SCHÉMA DE FONCTIONNEMENT (VRRPD DE BASE)



2.2 ÉTUDE

2.2.1 LOGICIELS QUI UTILISENT UN PRINCIPE SIMILAIRE.

- UCARP: (Une implémentation de CARP indépendante d'OpenBSD) Le changement d'état d'un serveur peut provoquer une action (lancement d'un script ou d'un binaire), mais UCARP ne fonctionne pas avec une Mac virtuelle, ce qui pose problème avec les équipements faisant du cache ARP. En cas de bascule, le temps de rétablissement doit aussi tenir compte de la mise à jour des caches ARP des hôtes du réseau, mise à jour dont le temps est variable et difficilement maîtrisable.

En ayant une MAC virtuelle qui bascule d'une machine à l'autre, il n'y aurait pas ce problème.

- VRRPD: Avec une Mac virtuelle, mais fonctionne sans scripts de changement d'état, de plus le serveur perd ses routes lors du changement d'adresse Mac (dans le code source, il démonte l'interface, change la MAC, et finalement remonte l'interface).

Il existe aussi keepalived basé sur le protocole VRRP, il est possible de lancer des scripts selon l'état du serveur comme dans UCARP, mais malheureusement il fonctionne sans MAC virtuelle.

2.2.2 LIMITATIONS DES LOGICIELS

Dans UCARP et VRRPD il n'est pas possible par défaut de surveiller l'état physique d'une machine complète ainsi que le bon fonctionnement des processus. Par exemple une interface débranchée ne provoque aucune action.

De plus les processus étant indépendants les uns des autres vous pouvez avoir des cartes avec des états différents (Par ex : eth0 = BACKUP et eth1 = MASTER)

2.3 MODIFICATIONS DE VRRPD

2.3.1 GESTION D'ÉTAT GLOBALE

Ces difficultés sont maintenant solutionnées par une gestion globale de l'état de la machine, les processus VRRPD se synchronisent entre eux sur le même serveur. Ils communiquent leur statut et informent les autres d'un éventuel dysfonctionnement.

Une machine ne peut donc plus fonctionner avec des états différents simultanément sur plusieurs interfaces.

2.3.2 LISTE DES MODIFICATIONS

J'ai ajouté trois commandes au programme vrrpd (-U -D sont équivalentes aux commandes du logiciel UCARP)

Usage: vrrpd -i ifname -v vrid [-f piddir] [-s] [-a auth] [-p prio] [-nh] ipaddr

-M : (-M x) Monitoring process and Network (Max 9) bascule automatiquement tous les processus VRRPD sur le MASTER en STATE BACKUP en cas de dysfonctionnement sur une carte (processus VRRPD manquant, link de la carte réseau tombé, etc ...). Cette commande synchronise l'état des processus et analyse le fonctionnement complet de la machine.

Avec U et D il est maintenant possible de passer une commande lors d'un changement d'état VRRP, elle peut tout simplement lancer un script (contenant des routes ou/et relancer un service par exemple)

-U : (-U): run to become a master) -> Script à exécuter lors du passage en MASTER

-D : (-D): run to become a backup) -> Script à exécuter lors du passage en BACKUP

Exemple :

```
./vrrpd -i eth0 -v 51 17.16.1.200 -M 2 -U /etc/scripts/MASTER.sh -D /etc/scripts/DOWN.sh
```

Client atropos : J'ai ajouté un client (atropos) permettant de visualiser et de modifier l'état d'une machine. Utilisable, par exemple, dans un script de supervision qui lors de la détection d'une anomalie bascule intégralement la machine en state backup.

3 EXPLOITATION

3.1 COMPILATION DU BINAIRE

Nécessite: libc6-dev pour la compilation

Installation:

1. Décompresser le fichier source
2. cd dans le répertoire
4. make clean
5. make
6. copier vrrpd dans votre path (e.g /usr/sbin)

Il existe aussi un script rudimentaire de compilation et d'installation (install.sh)

3.2 COMMANDES VRRPD/ATROPOS

vrrpd -i ifname -v vrid [-M monitor] [-s] [-a auth] [-p prio] [-nh] ipaddr

-h : display this short inlined help

-n : Dont handle the virtual mac address

-i ifname: the interface name to run on

-v vrid : the id of the virtual server [1-255]

-s : Switch the preemption mode (Enabled by default)

-a auth : (not yet implemented) set the authentication type

auth=(none|pass/hexkey|ah/hexkey) hexkey=0x[0-9a-fA-F]+

-p prio : Set the priority of this host in the virtual server (df: 100)

-d delay : Set the advertisement interval (in sec) (df: 1)

ipaddr : the ip address(es) of the virtual server

-U : (-U <file>): run <file> to become a master)

-D : (-D <file>): run <file> to become a backup)

-M : (-M x) Monitoring process and Network (Max 9)

Exemple de lancement sur la machine 1:

```
./vrrpd -i eth0 -v 51 ipaddr 10.16.1.200 -M 2 -U /etc/scripts/MASTER.sh -D /etc/scripts/DOWN.sh  
./vrrpd -i eth1 -v 52 ipaddr 10.17.1.200 -M 2 -U /etc/scripts/MASTER.sh -D /etc/scripts/DOWN.sh
```

Sur la machine 2:

```
./vrrpd -i eth0 -v 51 ipaddr 10.16.1.200 -M 2 -U /etc/scripts/MASTER.sh -D /etc/scripts/DOWN.sh  
./vrrpd -i eth1 -v 52 ipaddr 10.17.1.200 -M 2 -U /etc/scripts/MASTER.sh -D /etc/scripts/DOWN.sh
```

Explication:

Vrrpd fonctionne sur l'interface eth0 et eth1 dans les domaines virtuels 51 et 52, un domaine virtuel segmente vos réseaux Vrrpd. Ceci permet d'avoir plusieurs domaines de machines VRRPD sur un même lan sans aucune interférence.

M 2 - Signifie que chaque processus VRRPD connaît l'existence d'une autre instance sur la même machine et qu'il doit la superviser, il est possible de surveiller au maximum 9 processus vrrpd par machine.

U /etc/scripts/MASTER.sh - Signifie qu'en cas de passage en mode master vrrpd exécutera ce script

D /etc/scripts/DOWN.sh - Signifie qu'en cas de passage en mode backup vrrpd exécutera ce script

Atropos:

atropos --backup Be backup (caution: Don't use with priority !)

atropos --help This Page

atropos --state Status

atropos -backup → Provoque un basculement total en mode backup

atropos -state → Affiche l'état de la machine (Master ou backup)

Les scripts suivants permettent l'exploitation de la machine:

/etc/init.d/vrrp → Lance ou stop les processus VRRPD

/etc/scripts/vrrp_on.sh → Script de configuration de VRRPD

/etc/scripts/VRRP/Master.sh → Script exécuté lors du passage en mode Master

/etc/scripts/VRRP/Backup → Script exécuté en lors du passage en mode Backup

3.3 **SPECIFICITÉS**

3.3.1 **INTERFACE PHYSIQUE**

Il est parfois nécessaire d'avoir une interface manipulable facilement, par exemple le logiciel Openswan nécessite une interface pour créer ipsec0, malheureusement l'interface virtuelle de Vrrpd est « invisible ».

Dans ce type de cas il suffit de créer une interface alias avec la commande `ifconfig eth0:x` dans le script de changement d'état MASTER. Il ne faut surtout pas oublier d'ajouter aussi une ligne pour la démonter dans le script de changement d'état BACKUP afin d'éviter d'avoir deux IP totalement identiques sur le réseau suite à une bascule.

Il est donc nécessaire de bien réfléchir lors de la mise en place de votre domaine Vrrpd, dans certains cas il est préférable de mettre en place une adresse inutilisée sur l'interface virtuelle (`ipaddr`), et de mettre les véritables adresses de productions dans les scripts Master/Backup, dans tous les cas la mac virtuelle basculera bien d'une machine à l'autre pour toutes vos adresses.

Lors des bascules Vrrpd envoie systématiquement des gratuitous ARP sur toutes les interfaces de la machine afin d'éviter d'éventuels problèmes avec le cache ARP des équipements réseaux.

Informations diverses:

Quand l'état de Vrrpd n'est pas stable:

Il est parfois nécessaire de désactiver le spanning tree (aussi appelé STP) sur les ports de certains switch accueillant les machines en VRRP.

Idem pour la négociation de vitesse des ports sur l'équipement, je conseil plutôt de la fixer.

Comment avoir plusieurs IP virtuelles :

Vrrpd change la mac de l'interface il est donc simple d'avoir plusieurs IP qui basculent avec cette Mac

Pour cela il suffit d'utiliser les scripts MASTER/BACKUP

Par exemple :

Dans le script Master -> `etc/scripts/VRRP/Master.sh`

```
ifconfig eth0:0 192.168.2.2 netmask 255.255.255.0 up
```

Dans le script Backup -> `/etc/scripts/VRRP/Backup`

```
ifconfig eth0:0 192.168.2.2 down
```

Remarque : Il est très important de ne pas oubliez le script Backup sous peine de se retrouver avec un conflit d'adresses

VPN

Avec le VRRPD modifié il est parfaitement possible de secourir un VPN (Openswan) à condition d'avoir exactement la même configuration sur la machine de secours.

Il suffit d'utiliser les scripts MASTER/BACKUP pour monter et démonter le tunnel ainsi que son interface liée

Dans ipsec.conf voir la ligne interfaces= "ipsec0=eth0 :0"

4.1.1 INTERFACE IPSEC0

Parmi les changements apportés par le noyau 2.6, il y a le support natif de l'IPSec, ce mode de fonctionnement fait disparaître l'interface virtuelle ipsec0 utilisée à travers le module "KLIPS".

Ce problème est lié à l'utilisation du kernel 2.6 et en particulier le code kernel "26sec" ou module "NETKEY" utilisé pour le traitement IPsec avec lequel Openswan travaille en standard.

Pour l'instant la seule solution satisfaisante que j'ai pu tester passe par la construction d'un module kernel KLIPS pour openswan et chargement de ce dernier avant le lancement d'ipsec

Après différents essais en regardant les documentations disponibles sur Internet, mais qui ne fonctionnaient malheureusement pas, voici une petite méthode que j'ai mise au point (juillet 2006) afin de créer une interface ipsec avec l'ancien module KLIPS.

Il est possible qu'il existe d'autres solutions sur des distributions différentes ou bien que le produit a maintenant évolué. Il est important de vérifier avant de vous lancer.

Il faut télécharger les sources sur le site d'openswan

Dans le paquet source openswan et le rep modobj26/

Dans tous les fichiers, il faut mettre des commentaires aux lignes

contenant `skb->nf_debug = 0;`

En C un commentaire pour une ligne = //

Donc // `skb->nf_debug = 0;`

Il faut maintenant installer le paquet kernel source et kernel-headers

Ensuite la compilation du module doit fonctionner avec la commande suivante

`make clean`

`make KERNELSRC=/lib/modules/2.6.12-18mdk/build module mininstall`

2.6.12-18mdk = votre version de noyau

Il faut ensuite gzippier le module (gzip ipsec.ko) qui devient ipsec.ko.gz

Le copier dans le bon répertoire module (au besoin le créer)

```
cp /lib/modules/2.6.12-18mdkcustom/kernel/net/ipsec/ipsec.ko.gz /lib/modules/2.6.12-18mdk/kernel/net/ipsec/ipsec.ko.gz
```

Ensuite modifier le fichier modules.dep du noyau

```
vi /lib/modules/2.6.12-18mdk/modules.dep
```

Et ajouter le chemin du module avec : à la fin

```
/lib/modules/2.6.12-18mdk/kernel/net/ipsec/ipsec.ko.gz:
```

Il est maintenant possible de monter le module avec la commande

```
modprobe ipsec
```

Dans /etc/modules.conf ajouter ipsec pour une bonne prise en compte au démarrage