RAID et LVM

Quand vous formatez un disque pour le système de fichiers Ext3, vous préparer un espace pour stocker vos données et votre système.

Les données ne disposent d'aucune protection contre la corruption autre que les sauvegardes que vous aurez pensée à réaliser régulièrement. Je vous renvoie vers mon article précédent présentant quelques techniques de sauvegarde[1].

Une approche couramment utilisée pour sécurisée les données entreposées dans un serveur est de recopier automatiquement et en temps réel entre plusieurs disques. Cette technologie est appelée RAID.

1.1 RAID

Le RAID[2] est une technologie d'agrégation de disques durs. Ainsi, il est possible de présenter une grappe de disques durs sous la forme d'un seul est unique disque « virtuel » au système d'exploitation.

Cette technologie est largement utilisée dans les serveurs car, selon la topologie de disques retenue, elle permets de mettre en ligne des volumes de données bien plus larges que la taille d'un disque et/ou de garantir la préservation des données même en cas de défaillance d'un disque. De plus, en répartissant les blocs de données d'un même fichier sur plusieurs disques (« stripes »), on accélère les performances d'accès.

Le RAID est disponible en version matérielle intégrée au BIOS de la carte mère ou sous la forme de carte fille. Ces extensions existent pour tout type de connectique disque (ATA, SCSI, SAS, SATA...). Comme toute l'intelligence du RAID est déléguée à une carte fille, cette couche de gestion des disques n'a pratiquement aucun impact sur les performances de l'OS lequel ne voit qu'un seul et unique disque « virtuel ».

Il existe sous Linux et Windows une implémentation logicielle du RAID. Elle reprend les mêmes fonctionnalités que la version matérielle mais grève légèrement les performances du système. De plus, la gestion du RAID étant réalisée par l'OS, la gestion des disques n'est pas totalement invisible.

1.1.1 Topologies de RAID

Le choix d'une topologie RAID dépend des objectifs auxquelles doit répondre le serveur. Selon que la priorité est donnée à la taille de l'espace disponible, à la haute-disponibilités des données ou aux performances. Plusieurs niveaux de RAID sont tombés en désuétude. Je vais résumer ci-dessous les niveaux les plus fréquemment utilisés.

Les différents types d'architecture RAID sont numérotés à partir de 0 et peuvent se combiner entre eux (on parlera alors de RAID 0+1, 1+0, etc.). Les types les plus couramment utilisés sont décris ci-dessous.

RAID 0 : volume agrégé par bandes

Le RAID 0, également connu sous le nom d'« entrelacement de disques » ou de « volume

1 http://www.synergeek.fr/2008/07/sauvegarder-ses-donnees-sous-debian-4-scripts-simples-mais-efficaces
2 http://en.wikipedia.org/wiki/Redundant_array_of_independent_disks

agrégé par bandes » (*striping* en anglais) est une configuration RAID permettant d'augmenter significativement les performances de la grappe en faisant travailler *n* disques durs en parallèle. Le défaut de cette solution est que la perte d'un seul disque entraîne la perte de toutes les données. Le RAID 0 n'apportant pas de redondance, tout l'espace disque disponible est utile.

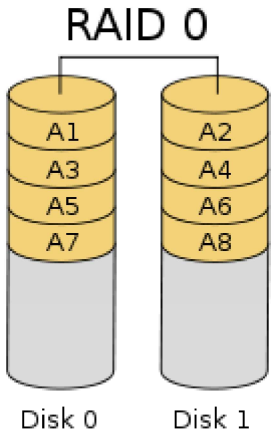

Illustration 1: RAID 0 : volume agrégé par bandes

RAID 1 : Disques en miroir

Le RAID 1 consiste en l'utilisation de disques redondants, chaque disque de la grappe contenant à tout moment exactement les mêmes données, d'où l'utilisation du mot « miroitage » (mirroring en anglais). La capacité totale est égale à celle du plus petit élément de la grappe. Cette solution offre un excellent niveau de protection des données. Les coûts de stockage sont élevés et directement proportionnels au nombre de miroirs utilisés alors que la capacité totale reste inchangée. Plus le nombre de miroirs est élevé, et plus la sécurité augmente, mais plus son coût devient prohibitif.

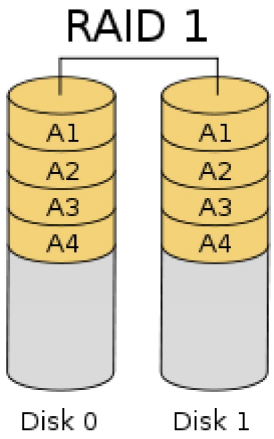

Illustration 2: RAID 1 : Disques en miroir

Les accès en lecture du système d'exploitation se font sur le disque le plus facilement accessible à ce moment-là. Les écritures sur la grappe se font de manière simultanée sur tous les disques, de façon à ce que n'importe quel disque soit interchangeable à tout moment.

Lors de la défaillance de l'un des disques, le contrôleur RAID désactive, de manière transparente pour l'accès aux données, le disque incriminé. Une fois le disque défectueux remplacé, le contrôleur RAID reconstitue, soit automatiquement, soit sur intervention manuelle, le miroir. Une fois la synchronisation effectuée, le RAID retrouve son niveau initial de redondance.

RAID 5 : volume agrégé par bandes à parité répartie

Le RAID 5 combine la méthode du volume agrégé par bandes (striping) à une parité répartie. La parité, qui est incluse avec chaque écriture se retrouve répartie circulairement sur les différents disques. Chaque bande est donc constituée de N blocs de données et d'un bloc de parité. Ainsi, en cas de défaillance de l'un des disques de la grappe, pour chaque bande il manquera soit un bloc de données soit le bloc de parité. Si c'est le bloc de parité, ce n'est pas grave, car aucune donnée ne manque. Si c'est un bloc de données, on peut deviner son contenu à partir des N-1 autres blocs de données et du bloc de parité. L'intégrité des données de chaque bande est préservée. Donc non seulement la grappe est toujours en état de fonctionner, mais il est de plus possible de reconstruire le disque une fois échangé à partir des données et des informations de parité contenues sur les autres disques.

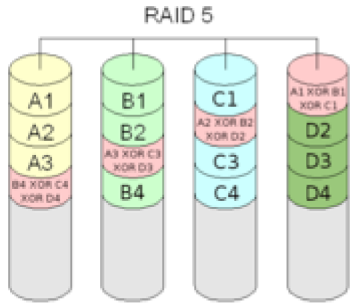

Illustration 3: RAID 5 : volume agrégé par bandes à parité répartie

On voit donc que le RAID 5 ne supporte la perte que d'un seul disque à la fois. Ce qui devient un problème depuis que les disques qui composent une grappe sont de plus en plus gros (1 To et plus). Le temps de reconstruction de la parité en cas de disque défaillant est allongé. Il est généralement de 2h pour des disques de 300 Go contre une dizaine d'heure pour 1 To. Pour limiter le risque il est courant de dédier un disque dit de spare. En régime normal il est inutilisé. En cas de panne d'un disque il prendra automatiquement la place du disque défaillant. Bien sûr pendant tout le temps du recalcul de la parité le disque est disponible normalement pour l'ordinateur qui se trouve juste un peu ralenti.
Ce système nécessite impérativement un minimum de trois disques durs. Ceux-ci doivent généralement être de même taille.
La capacité de stockage utile réelle, pour un système de X disques de capacité c identiques est de $(X - 1) * c$.
Ce système allie sécurité (grâce à la parité) et bonne disponibilité (grâce à la répartition de la parité), même en cas de défaillance d'un des périphériques de stockage.

RAID 6

Le RAID 6 est une évolution du RAID 5 qui accroît la sécurité en utilisant n informations redondantes au lieu d'une. Il peut donc résister à la défaillance de n disques.

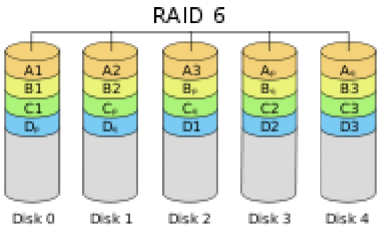

Illustration 4: RAID 6

Si la sécurité est plus grande, le coût en matériel est plus élevé et la vitesse est moindre. La puissance CPU nécessaire pour calculer les redondances et surtout pour reconstruire un volume défectueux est également nettement plus importante.

Les niveaux de RAID combinés

Fondamentalement, un niveau de RAID combiné est l'utilisation d'un concept de RAID classique sur des éléments constitutifs qui sont eux-mêmes le résultat d'un concept RAID classique. Le concept utilisé peut être le même ou différent.

La syntaxe est encore un peu floue mais on peut généralement considérer que le premier chiffre indique le niveau de raid des "grappes" et que le second indique le niveau de raid global. Dans l'absolu rien n'empêche d'imaginer des RAID combinés à 3 étages ou plus mais cela reste pour l'instant plus du domaine de la théorie et de l'expérimentation.

le RAID 01 (ou RAID 0+1)

Il permet d'obtenir du *mirroring* rapide puisqu'il est basé sur des grappes en stripping. Chaque grappe contenant au minimum 2 éléments, et un minimum de 2 grappes étant nécessaire, il faut au minimum 4 unités de stockage pour créer un volume RAID0+1.

La fiabilité est moyenne car un disque défectueux entraîne le défaut de toute une grappe. Par ailleurs, cela allonge beaucoup le temps de reconstruction et dégrade les performances pendant la reconstruction. L'intérêt principal est que dans le cas d'un miroir à 3 grappes ou plus, le retrait volontaire d'une grappe entière permet d'avoir une sauvegarde "instantanée" sans perdre la redondance.

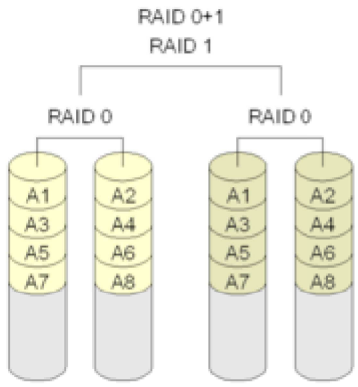

Illustration 5: le RAID 01 (ou RAID 0+1)

RAID 10 (ou RAID 1+0)

Il permet d'obtenir un volume agrégé par bande fiable (puisqu'il est basé sur des grappes répliquées). Chaque grappe contenant au minimum 2 éléments et un minimum de 2 grappes

étant nécessaire, il faut au minimum 4 unités de stockage pour créer un volume RAID10.

Sa fiabilité est assez grande puisqu'il faut que tous les éléments d'une grappe soient défectueux pour entraîner un défaut global. La reconstruction est assez performante puisqu'elle ne mobilise que les disques d'une seule grappe et non la totalité.

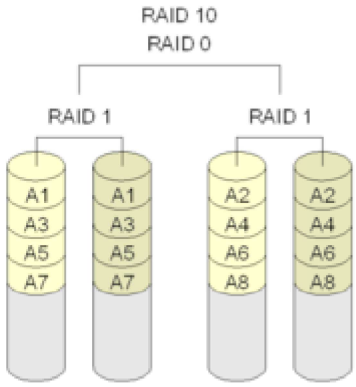

Illustration 6: RAID 10 (ou RAID 1+0)

RAID 50

Il permet d'obtenir un volume agrégé par bandes basé sur du RAID 5. Chaque grappe contenant au minimum 3 disques, et un minimum de 2 grappes étant nécessaire, il faut au minimum 6 unités de stockage pour créer un volume RAID 50. Un des meilleurs compromis lorsque l'on cherche la rapidité sans pour autant vouloir trop dégrader la fiabilité. En effet, l'agrégat par bande (fragile) repose sur des grappes redondantes. Il suffit cependant que 2 disques d'une même grappe tombent en panne pour le mettre en défaut.

C’est une erreur fréquente que de confondre le RAID avec une sauvegarde. Certes, le RAID peux vous garantir contre la panne d'un ou plusieurs disques mais:

- il n'offre aucun historique de données,
- il ne vous garantie pas contre une panne du PC!

1.1.2 Installation

Le RAID logiciel sous Linux est géré « mdadm »[3]. Sous Debian Lenny, cet outil est installé par la commande (en tant que l'utilisateur « root »):

```
apt-get install -y mdadm
```

Création d'un RAID5

Dans le cadre de cet article, nous considérerons que le PC est équipé de quatre disques. */dev/sda* contient le système, */dev/sdb*, */dev/sdc* et */dev/sdd* sont destinés au stockage des données sous la forme d'un RAID5 nommé */dev/md0*. Cet espace de stockage sera monté sous */media/raid5*.

La première étape consiste à créer sur chacun des disques destinés au RAID une partition de type « FD » ou « Linux RAID ». Pour cela, utilisez l’outil « cfdisk ». Ainsi, pour le disque /dev/sdb, taper la commande:

3 http://en.wikipedia.org/wiki/Mdadm

```
cfdisk /dev/sdb
```

Depuis l'interface de « cfdisk », choisissez le menu « New ».

Illustration 7: Interface de gestion des partitions avec cfdisk

Vous souhaitez créer une partition «PRIMAIRE ».

Vous pouvez valider la taille indiquée par défaut car nous allons utiliser l'intégralité du disque pour le RAID.

Votre partition est créée. Maintenant, vous devez indiquez le type de données que contiendra cette partition. Pour cela, choisissez le menu « TYPE ».

Saisissez le type correspondant au RAID logiciel c'est à dire « FD ».

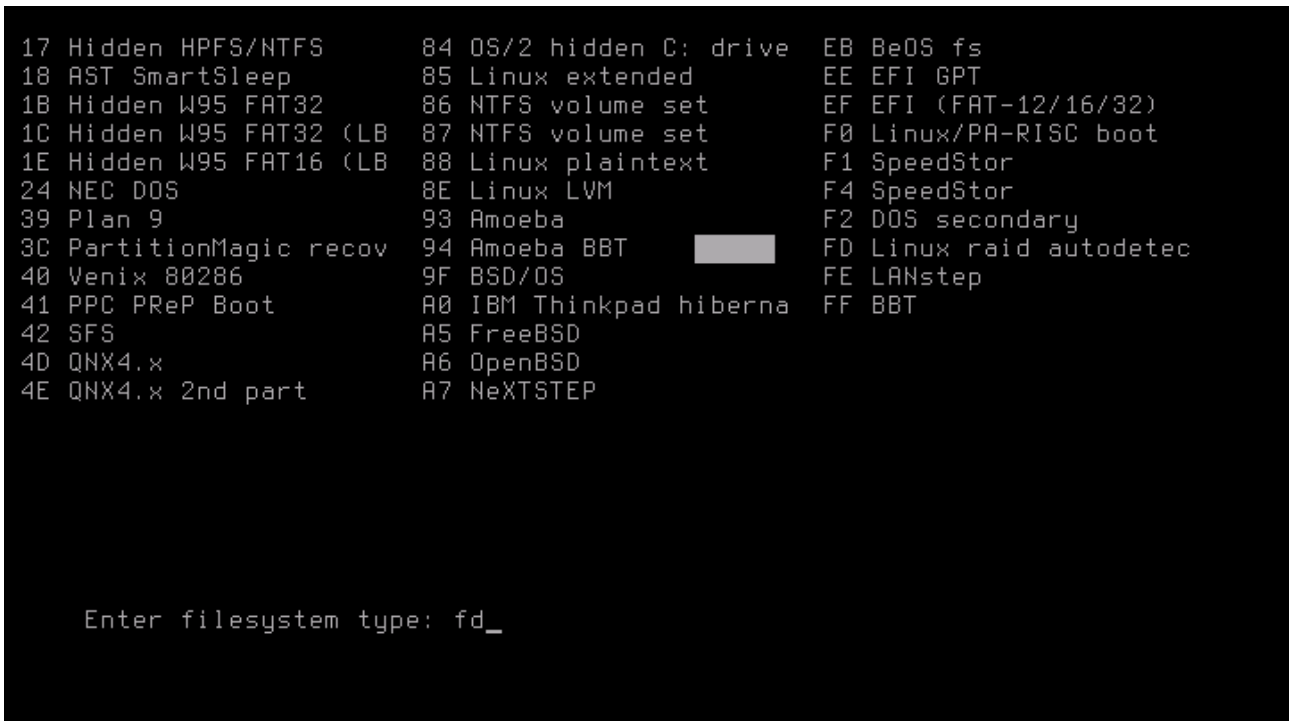

Illustration 8: Choix du type de partition disque

Il ne vous reste plus qu'à appliquer votre choix au disque avec le menu « WRITE ».

Vous pouvez désormais quitter le programme « cfdisk » en utilisant le menu « QUIT ».

Répétez cette opération pour tous les disques « /dev/sdc » et « /dev/sdd ».

```
cfdisk /dev/sdc
cfdisk /dev/sdd
```

La création du RAID est réalisée par la commande:

```
mdadm –create –verbose /dev/md0 –level=5 –raid-devices=3
/dev/sdb /dev/sdc /dev/sdd
```

Sur un RAID logiciel de si petite taille, la création est quasiment instantanée. Mais, avec des disques de taille plus conséquente ou dans le cas de la reconstruction d'un RAID existant, cette opération peut demander plusieurs heures. La commande ci-dessous permet de suivre l'avancement de la création du RAID:

```
cat /proc/mdstat
```

```
debian:~# cat /proc/mdstat
Personalities : [linear] [multipath] [raid0] [raid1] [raid6] [raid5] [raid4] [ra
id10]
md0 : active (auto-read-only) raid5 sdd1[3](S) sdc1[1] sdb1[0]
      16771584 blocks level 5, 64k chunk, algorithm 2 [3/2] [UU_]

unused devices: <none>
debian:~# _
```

Illustration 9: Etat de création d'un Raid

La commande ci-dessous vous informera de la structure et de la sante de votre RAID:

```
mdadm -detail /dev/md0
```

```
192.168.1.126 - KiTTY
  Used Dev Size : 8385792 (8.00 GiB 8.59 GB)
   Raid Devices : 3
  Total Devices : 3
Preferred Minor : 0
    Persistence : Superblock is persistent

    Update Time : Fri Mar 20 14:01:40 2009
          State : clean
 Active Devices : 3
Working Devices : 3
 Failed Devices : 0
  Spare Devices : 0

         Layout : left-symmetric
     Chunk Size : 64K

           UUID : c3d44f99:57e4030d:9d4deba6:47ca997f
         Events : 0.10

    Number   Major   Minor   RaidDevice State
       0       8       17        0      active sync   /dev/sdb1
       1       8       33        1      active sync   /dev/sdc1
       2       8       49        2      active sync   /dev/sdd1
(END)
```

Illustration 10: Statut d'un Raid

Une fois le périphérique « /dev/md0 » crée, vous pouvez le formater en Ext3 comme tout disque physique.

```
mkfs.ext3 -j /dev/md0
```

1.1 Configuration du gestionnaire de RAID

« mdadm » installe un agent qui, s'il est configuré, va suivre l'état de vos RAID et réaliser des opérations automatisées en cas de défaillance. Dans le cadre de cet article, nous allons simplement demander à recevoir un email. Mais on pourrait le programmer pour utiliser un disque « spare » en cas de défaillance d'un des éléments du RAID.

La commande suivante va générer automatiquement le fichier de configuration du service de démarrage du RAID.

```
mdadm --examine --scan >> /etc/mdadm/mdadm.conf
```

Cette manipulation souffre d'un léger BUG sans conséquence si ce n'est un message d'erreur dans vos journaux de démarrage. Il suffit d'éditer le fichier « /etc/mdadm/mdadm.conf » afin de remplacer la chaine « 00.90 » par « 0.90 ».

```
nano /etc/mdadm/madm.conf
```

Ajoutez en fin de fichier la ligne MAILADDR= xxx@yyy.zz où xxx@yyy.zz est l'adresse email vers laquelle vous souhaitez recevoir les alertes de l'agent RAID.

1.2 Configurer Debian pour émettre des alertes emails

En cas de défaillance d’un disque du RAID, il est fort probable que votre serveur ne sache pas comment vous envoyer un email d’alerte. Debian est normalement équipée du serveur de messagerie « exim4 »[4] mais celui-ci renvoie les messages vers la boite locale de « root ».

Vous allez configurer « exim » afin qu'il envoie tous les mails vers l'internet. Pour cela, vous vous appuierez sur le serveur SMTP de Google Gmail. L'utilisation de ce serveur SMTP public nécessite de posséder un compte Gmail (gratuit!). L'adresse email du destinataire des alertes ne souffre par contre qu’aucune limitation.

Tout d'abord assurez-vous que « exim » a bien été installé avec votre distribution. Pour cela, tapez la commande suivante:

```
dpkg -l |grep exim
```

Si « exim » est disponible, vous devriez obtenir une réponse similaire à celle-ci:

4 http://fr.wikipedia.org/wiki/Exim

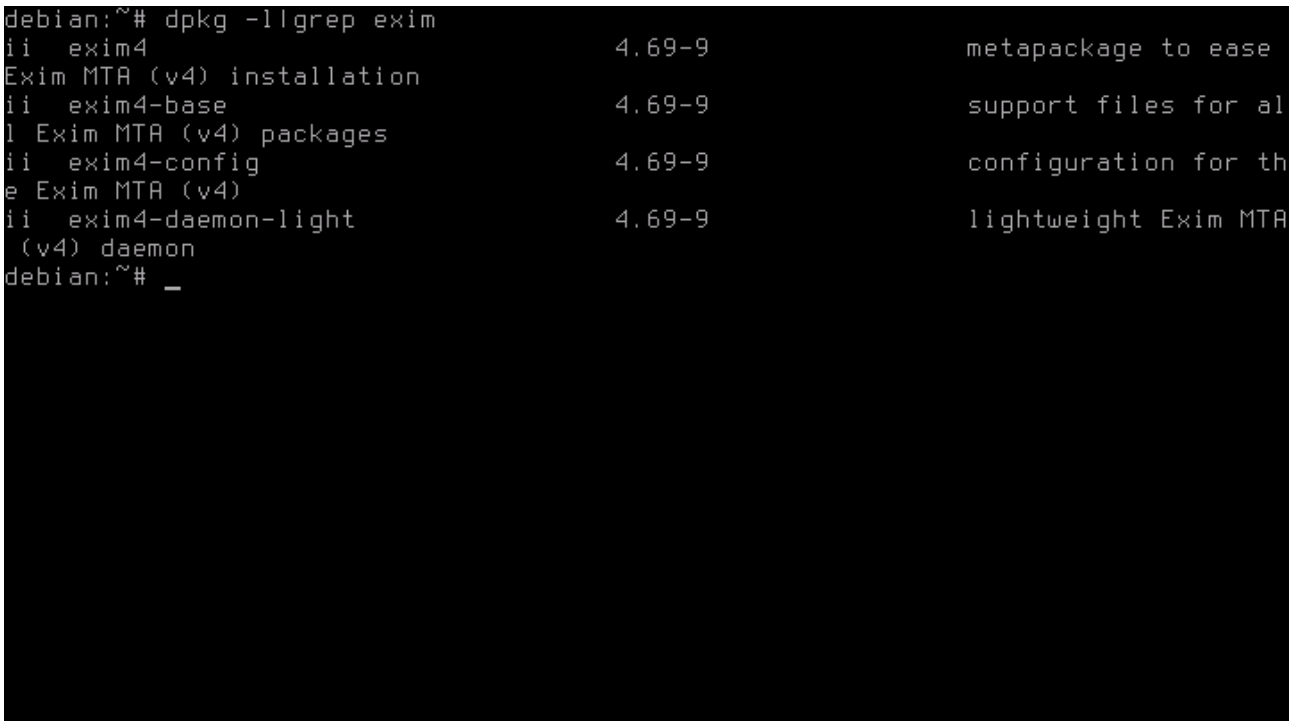

Illustration 11: Listing des paquets Debian installés pour "exim"

Si ce n'est pas le cas, installez « exim » avec la commande ci-dessous mais pensez à vérifier avant que d'autres serveurs comme « postfix » ne sont pas déjà présents .

```
apt-get install -y exim4
```

Vous allez ensuite relancer la configuration de « exim ». Pour cela, tapez la commande:

```
dpkg-reconfigure exim4-config
```

Validez pour passer la page d'introduction.

Choisissiez, sur l'écran suivant, le troisième choix c'est à dire que tous les emails sortants seront redirigés vers un serveur SMTP externe (« smarthost »).

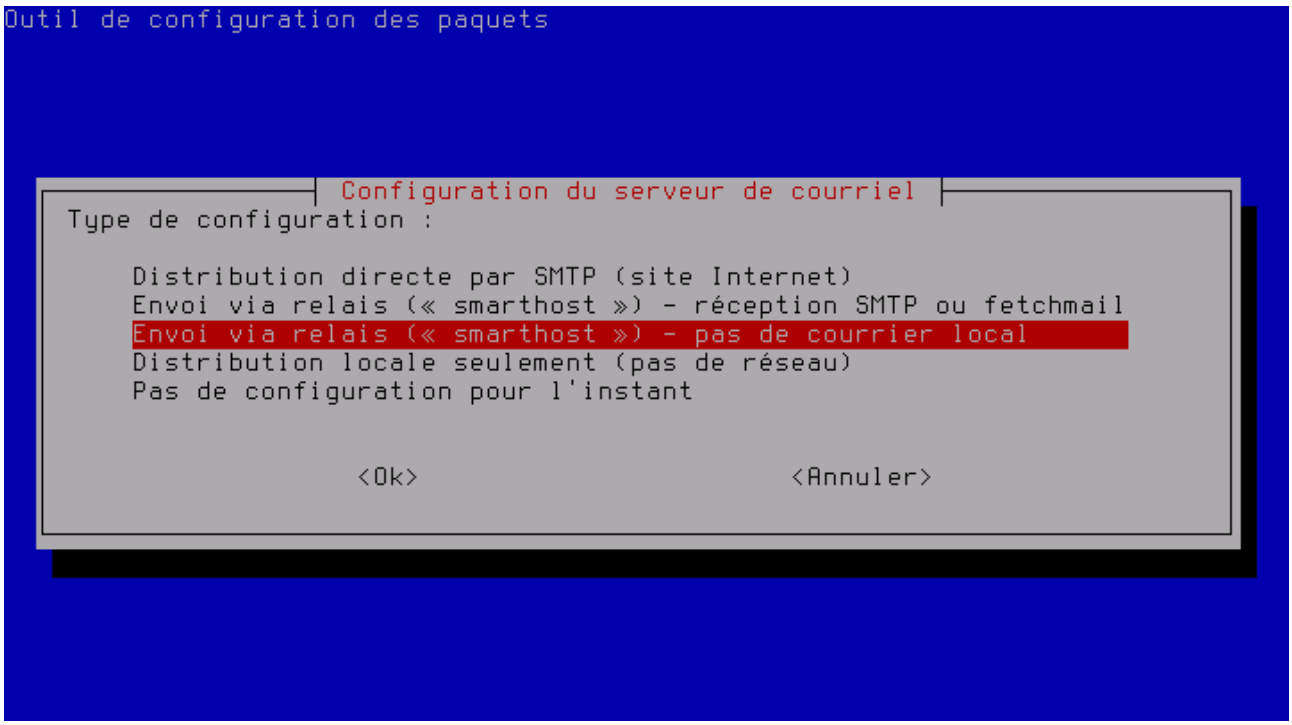

Illustration 12: Choisir le type de distribution SMTP pour "exim"

Indiquez sur l'écran suivant un identifiant pour votre serveur. Cet identifiant doit être une adresse email conforme c'est à dire de la forme xxx@yyy.zz mais non nécessairement correspondre à une véritable boite email.

Passez l'écran suivant.

Le serveur « exim » écoutera les requêtes qui lui sont faites sur l'interface « loopback » du PC soit l'adresse IP « 127.0.0.1 ».

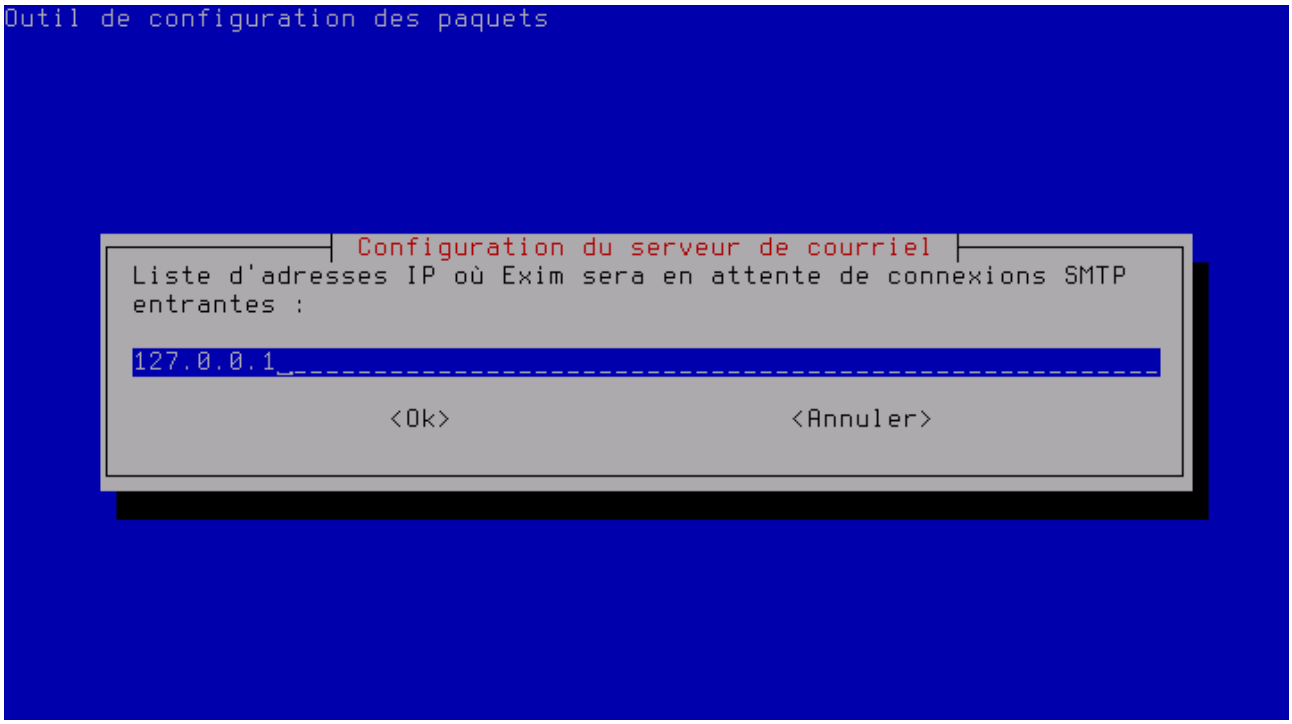

Illustration 13: Choix de l'adresse IP d'écoute de "exim"

Comme ce serveur « exim » n'a pas vocation a traiter des emails issus d'autres serveurs que le PC local, n'indiquez rien dans le champ « autres destinations.... ».

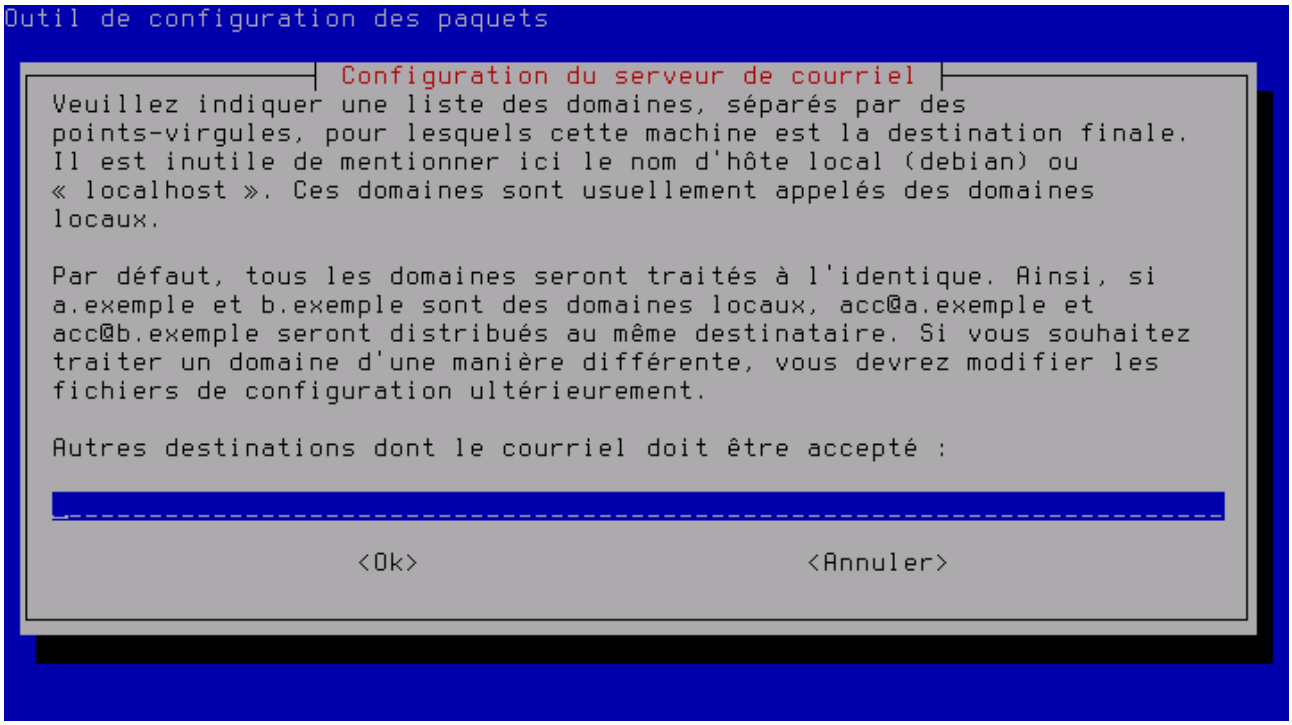

Illustration 14: Choix des domaines gérés par "exim"

Ne saisissez rien dans le champ suivant.

L'étape suivante permet d'indiquer le serveur SMTP vers lequel les emails seront redirigés. Indiquez ici l'adresse du serveur SMTP de Gmail. Comme les transactions avec ce serveur sont sécurisées, le port 587 est utilisé.

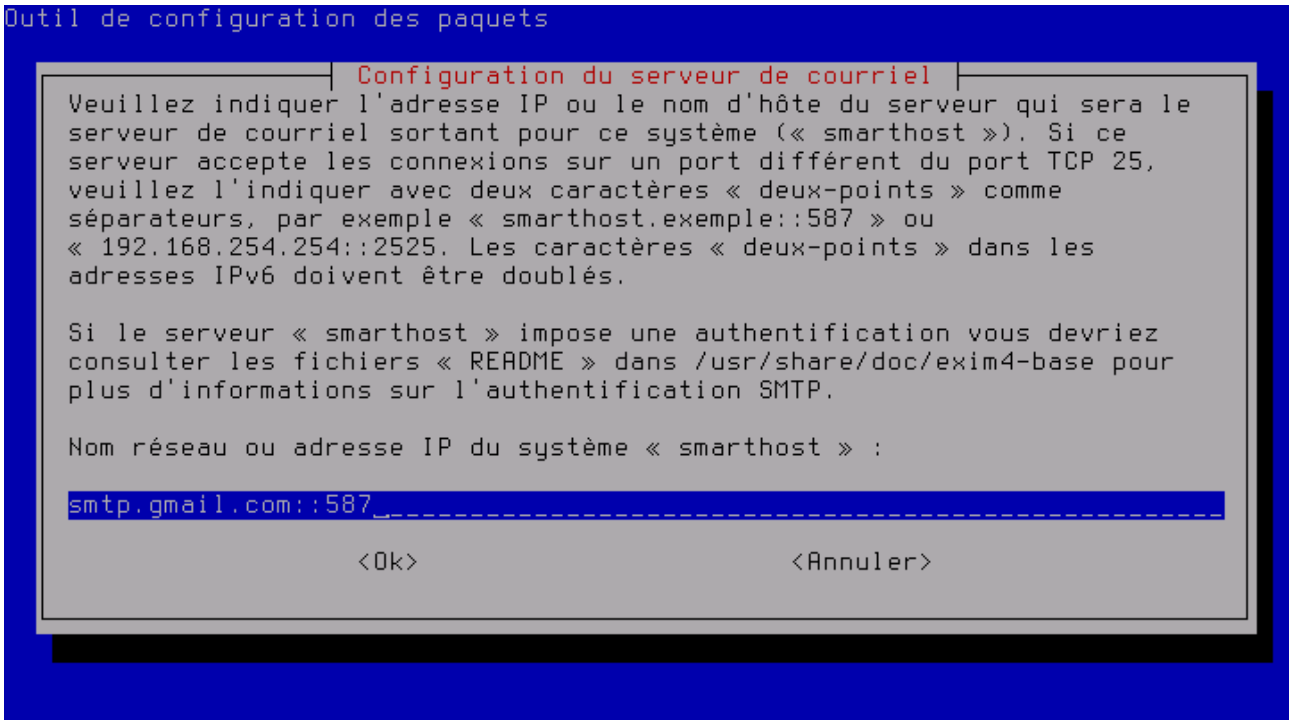

Illustration 15: utilisez le serveur SMTP de Gmail comme relai pour "exim"

Validez l'écran suivant.

Comme vous ne configurez pas « exim » de façon complexe, inutile de diviser les fichiers de configurations

La configuration du fonctionnement de « exim » est terminée. Toutefois, comme le serveur de Gmail requière une authentification avant d'accepter de relayer des mails, vous allez devoir éditer le fichier « /etc/exim4/passwd.client » pour y indiquer vos identifiants Gmail.

```
Nano /etc/exim4/passwd.client
```

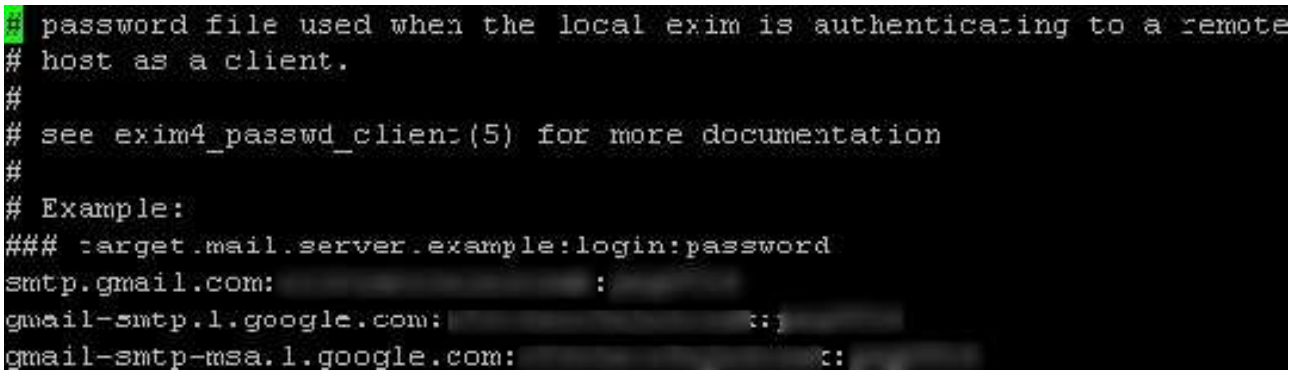

Illustration 16: Configuration du relai "exim" vers Gmail

Vous pouvez associer une adresse email à chacun de vos comptes locaux sur Debian.; Ainsi, tous les messages systèmes seront automatiquement relayés à Gmail à destination de l'utilisateur. Pour cela éditez le fichier « /etc/email-adresses ».

```
nano /etc/email-addresses
```

```
# This is /etc/email-addresses. It is part of the exim
package
#
# This file contains email addresses to use for outgoing
mail. Any local
# part not in here will be qualified by the system domain as
normal.
#
# It should contain lines of the form:
#
#user: someone@isp.com
#otheruser: someoneelse@anotherisp.com
```

```
root: xxxx@yyy.zzz
user1: xxxx@yyy.zzz
user2: xxxx@yyy.zzz
```
Listing1: Contenu de « /etc/email-addresses » pour rediriger les alertes

Il ne reste plus qu'à relancer « exim » afin d'appliquer les changements réalisés.

```
/etc/init.d/exim4 restart
```
Afin de valider, le bon fonctionnement de « email », tapez la commande suivante:

```
mail xxx@yyy.zzz
```
Saisissez un sujet, le corps du message (CTRL+D pour terminer). Vous devriez recevoir un email dans la boîte xxx@yyy.zz après quelques secondes.

1.3 Montage automatique du Raid

Si le fichier « /etc/mdadm/mdadm.conf » a correctement été configuré et que le point de montage du RAID est déclaré dans le fichier « /etc/fstab », le RAID devraient être démarrés automatiquement avec le système.

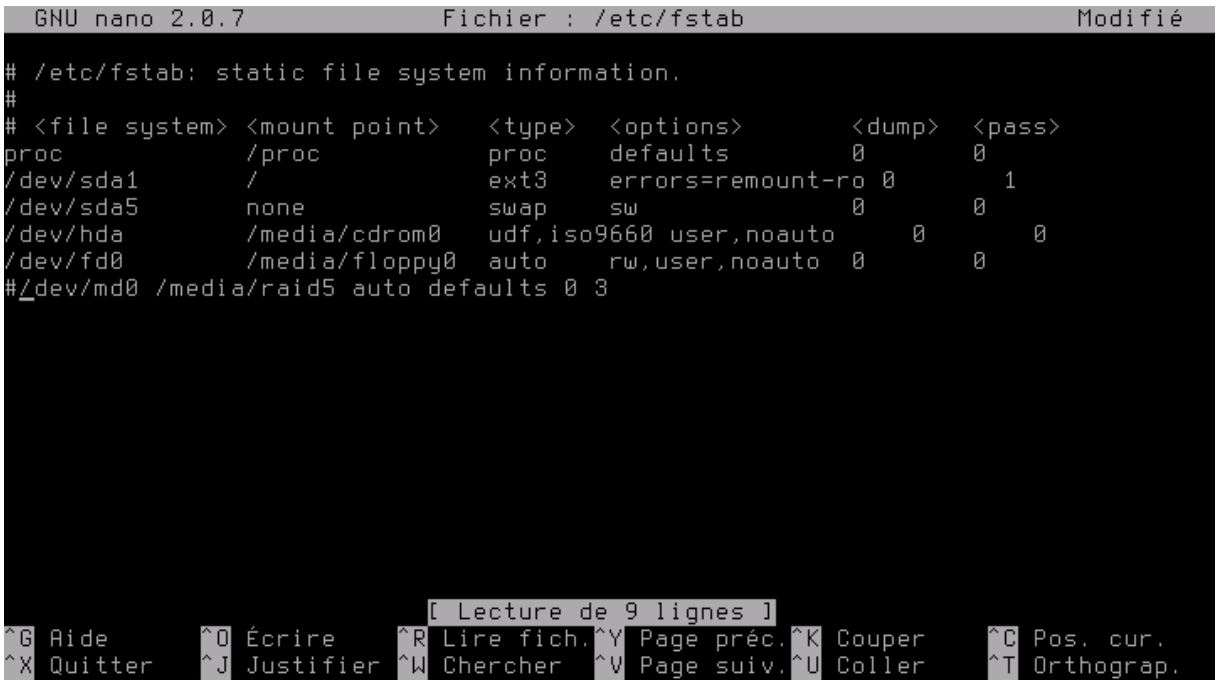

Illustration 17: Coinfiguration du RAID au démarrage depuis "/etc/fstab"

Ainsi, l'espace de stockage est immédiatement opérationnel!

1.4 Gestion de la défaillance d'un disque

Imaginons que le disque « /dev/sdb » soit défaillant.

Si la notification par email fonctionne, vous allez recevoir un email comme celui-ci:

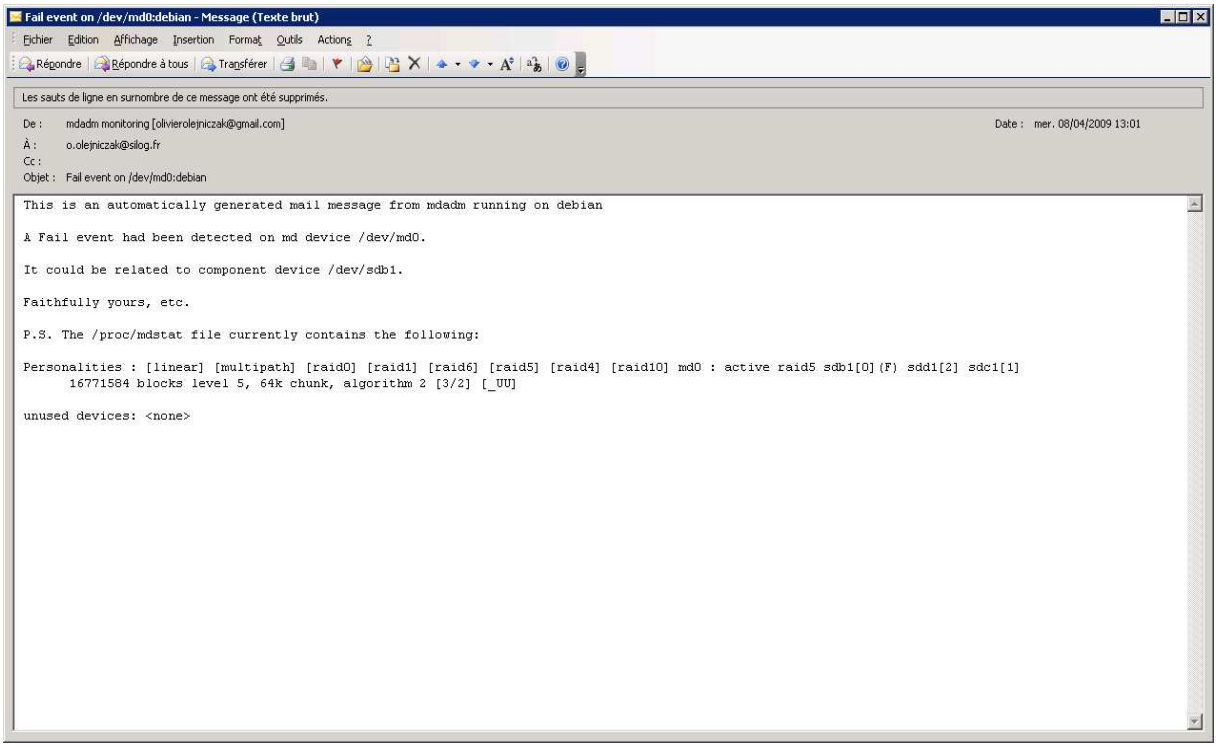

Illustration 18: Email d'alerte en cas de défaillance du RAID

Vous devez agir et tout d'abord informer le RAID de la défaillance du disque. Pour cela, tapez la commande suivante:

```
mdadm –manage –-set-faulty /dev/md0 /dev/sdb1
```

Retirez ensuite le disque du RAID.

```
mdadm –manage --remove /dev/md0 /dev/sdb1
```

Remplacer physiquement le disque dans le PC puis informez le RAID de la présence du nouveau disque.

```
mdadm –manage --add /dev/md0 /dev/sdb1
```

Vous pouvez suivre la reconstruction du RAID avec la commande:

```
mdadm –detail /dev/md0
```

Les données restent disponibles pendant la période de reconstruction.