Infusion

DRBD et haute disponibilité

André Schaaff

5 avril 2013

Centre de Données
astronomiques de Strasbourg

DRBD, c'est quoi ?

- Distributed Replicated Block Device
- DRBD peut être considéré comme étant un RAID 1 réseau
- Il permet de créer un miroir entre 1 espace disque E1 d'une machine Primaire et 1 espace disque E2 d'une machine Secondaire
- Seul E1 est visible tant que la machine Primaire est opérationnelle
- Dans l'absolu on peut faire un RAID réseau entre 2 machines peu importe leur localisation, pourvu que le réseau soit d'une qualité suffisante.
- Commandes de gestion de reconstruction en cas de problème

Il manque quelque chose…

- DRBD ne suffit pas car il faut gérer également le basculement en cas de panne !
- On peut utiliser Linux HA qui offre un mécanisme de heartbeat qui permettra, en cas de défaillance de la machine Primaire le basculement vers la machine Secondaire.
- Finalement on aboutit à un « système RAID » qui permet « d'encaisser » la perte d'une machine alors qu'un RAID classique offre uniquement une tolérance aux défaillances de disques.
- DRBD est compatible Pacemaker, heartbeat, Corosync, etc.

Comment se fait la synchronisation ?

- DRBD offre trois modes de fonctionnement appelés Protocole A, B et C
- La validation de la transaction dépendra du protocole
- La tolérance aux pannes sera également variable.
- Détaillons ces 3 modes

Le mode asynchrone (Protocole A)

- Il apporte le niveau de tolérance le moins élevé
  - La validation de la transaction intervient dès que les données sont copiées sur la machine Primaire et en cours d'envoi vers la machine Secondaire
  - Il y a donc un risque de perte de données si une panne de la machine Primaire survient alors que les données ne sont pas encore transmises (ou pas correctement) à la machine Secondaire
  - C'est un mode utilisé lorsque les 2 machines sont séparées par une grande distance géographique (latence) ou reliées par des réseaux peu performants
  - Performance maximale au niveau de la machine Primaire car pas de temps d'attente

Le mode semi-synchrone (Protocole B)

- C’est le mode intermédiaire, tolérance / performances
  - La transaction est considérée comme validée dès que les données sont écrites sur la machine Primaire et que les paquets correspondants sont arrivés sur la machine Secondaire mais l’écriture réelle des données sur la machine Secondaire n’est pas certaine au moment de la validation de la transaction
  - Il y a donc un risque de perte de données si un problème survient sur la machine Secondaire avant que les données ne soient réellement écrites

Le mode synchrone (Protocole C)

- C’est le mode le plus tolérant aux pannes
  - La transaction est validée lorsque l’écriture des données est terminée à la fois sur les machines Primaire et Secondaire
  - Les performances de ce mode dépendent du lien entre les 2 machines
  - Les données ne sont perdues qu’en cas de destruction simultanée des 2 machines
  - C’est le mode utilisé au CDS et c’est le mode le plus utilisé pour DRBD

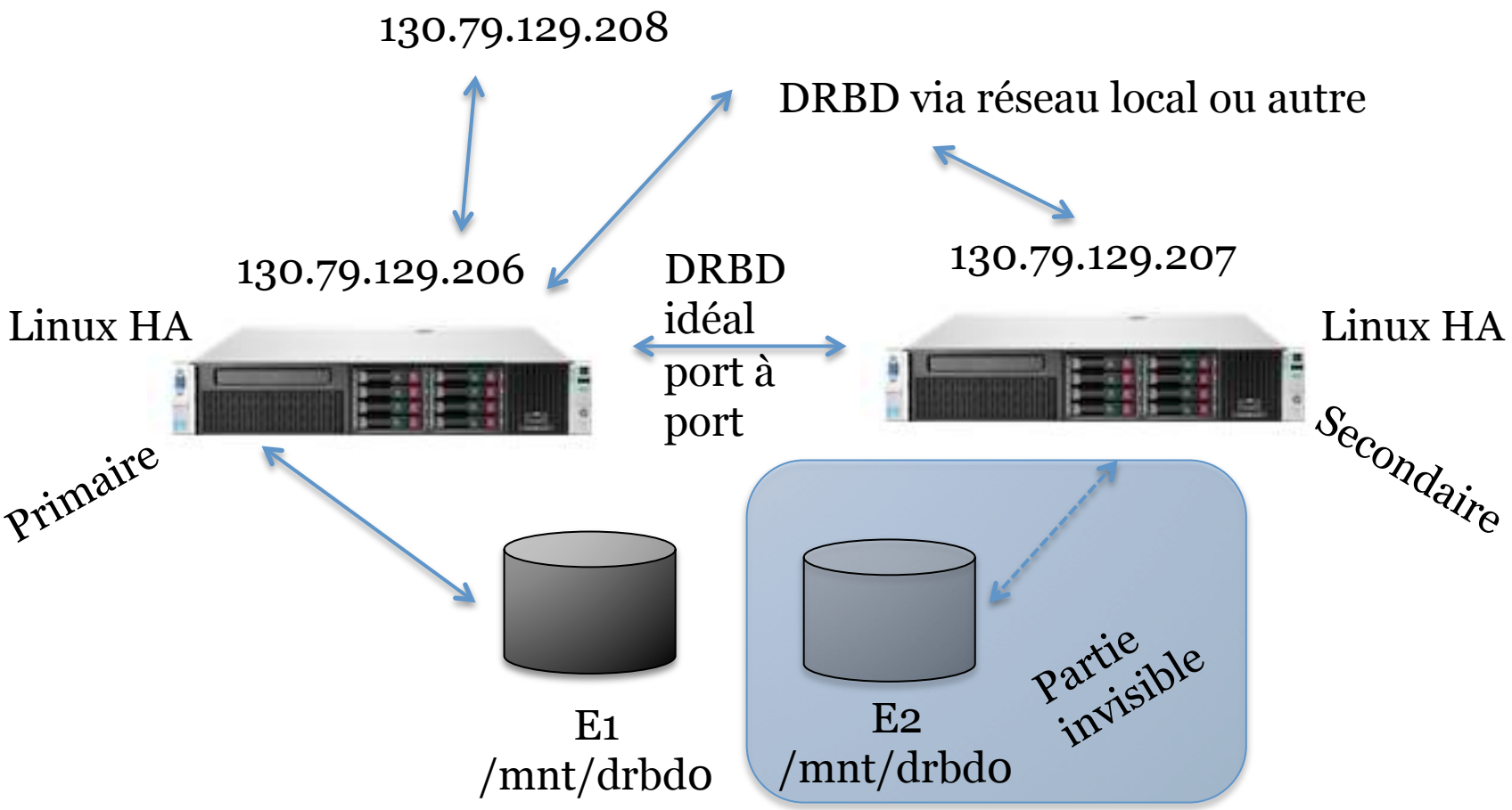
Au CDS

- 2 serveurs HP (*.206 et *.207) avec la configuration unitaire suivante
  - Système : 2 disques de 175Mo en RAID 1
  - Données : 6 disques de 1 To en RAID 5
- La partie données est gérée via DRBD
  - /mnt/drbd0
- Linux HA pour la partie surveillance / basculement
- Une adresse IP virtuelle (*.208) qui bascule entre les 2 serveurs en cas de problème

Un peu de concret

- En 4 ans, un seul problème de « désynchronisation » réglé en quelques heures (temps mis par la reconstruction)
- Un petit schéma de l'architecture…

Architecture



La main à la pâte…

- Installation de drbd (drbd8-utils)
- On remplit le fichier de configuration /etc/drbd.conf
  - Définition des nœuds, de la partition, etc.
- DRBD offre un ensemble de commandes qui permettent de faire toutes manipulations nécessaires (démarrage synchronisation, demande de la première copie bloc à bloc, etc.).
- On met en place la partie basculement avec l'outil que l'on préfère
- On simule des pannes pour vérifier que le basculement est opérationnel
- Etc.

Exemple de simulation de panne

- On éteint ou on coupe le réseau de la machine Primaire
- Le mécanisme de heartbeat détecte le problème et bascule la machine secondaire en Primaire ce qui a pour effet de rendre sa partition DRBD visible
- Lorsque la machine « anciennement » Primaire est remise en ligne, celle-ci se re-synchronise avec la machine qui a pris le relais mais se retrouve en machine Secondaire
- On peut évidemment la rebasculer en Primaire

A noter

- Nous avons vu le mode Actif / Passif avec donc une machine de secours en « attente »
  - On peut utiliser n'importe quel système de fichiers
- Il est existe également un mode Actif / Actif
  - Il faut utiliser des systèmes de fichiers comme GFS ou OCFS
  - On ajoute dans le fichier drbd.conf une instruction du genre allow-two-primaries
- Les développeurs de DRBD prévoient une gestion multi-nœuds à partir de la version 9.
- Il existe un outil visuel de monitoring.

Discussion

- Questions ?
- Commentaires ?
- On commençait à s’endormir et on passe à la suivante ?

Conclusion

- Au début l'utilisation de ce genre de technique peut faire peur
  - La synchronisation des blocs est-elle fiable ? Ne vais-je pas perdre mes données ? Est-ce bien raisonnable de faire cela avec des volumes de cette importance (il y a 4 ans …) ?
- Pas très rassuré lors du lancement de la commande manuelle lors du seul problème de désynchronisation
- Finalement, cela c'est plutôt bien passé !
- Il serait sans doute intéressant de tester le mode Actif / Actif

Liens

- La référence : http://www.drbd.org/
- Linux HA, http://www.linux-ha.org/
- http://doc.ubuntu-fr.org/heartbeat
- http://www.unixgarden.com/index.php/gnu-linux-magazine-hs/drbd-la-replication-des-blocs-disque
- http://fr.wikipedia.org/wiki/DRBD
- http://doc.ubuntu-fr.org/tutoriel/mirroring_sur_deux_serveurs
- Etc.