tetaneutral.net - Evolution #16

Agrégation de lien

18/07/2011 09:51 - Laurent GUERBY

Statut:	Fermé	**Début:**	18/07/2011
Priorité:	Normal	**Echéance:**	
Assigné à:	Laurent GUERBY	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0.00 heure
Version cible:			

Description

Les solutions actuelles d'agregation :

- bonding
- MLPPP
- iptables mark random

En pratique on souhaite un peu plus :
- equilibrage dynamique sur liens dissymetriques ADSL+wifi (cas de trebons.net)
- masquage de perte lien wifi pour debit TCP ~= UDP,

Prototype en python de pilote tap en cours par Laurent.

Historique

#1 - 22/08/2011 09:30 - Laurent GUERBY

Quelques liens :

http://www.secdev.org/projects/tuntap_udp/
Tunneling over UDP using tun/tap in Python or C
Here are two programs demonstration programs that use tun/tap to do IP or Ethernet tunneling over UDP. One is written in python : tunproxy.py and the other in C : tunproxy.c.

=> le programme python est une tres bonne base, sur un Atom N270 @ 1.6 GHz on obtiens 80 Mbit/s effectif iperf sur un lien 100 Mbit/s donc ok pour prototypage et pour ADSL.

http://alex.king.net.nz/tuntap.html
Python TUN/TAP Module for Linux

#2 - 22/08/2011 19:09 - Laurent GUERBY

" masquage de perte lien wifi pour debit TCP ~= UDP,"

Sur un lien wifi en pratique il y a toujours une perte
entre un iperf TCP et iperf UDP. Par exemple sur notre
lien de 3.7km on a 80-100 Mbit/s mesurés en iperf UDP
mais en TCP on a plutot 40-60 Mbit/s.

La cause probable est les faibles mais non nulles pertes de paquet qui empechent TCP d'atteindre le maximum du lien, TCP interpretant de la perte comme etant de la congestion. Mais on n'est
pas tout a fait sur, peut-etre il y a moyen de rattraper
cette difference.

#3 - 16/09/2011 00:34 - Laurent GUERBY

From: vincent daligault <ligro.daligro@gmail.com>
Reply-to: fai-locaux@fdn.fr
To: fai-locaux <fai-locaux@fdn.fr>
Subject: Re: [fai-locaux] Rencontre RMLL : De l'échange autours de la fibre chez les FAI locaux ?
Date: Thu, 15 Sep 2011 23:29:45 +0200

Bonsoir,

Je réponds un peu en retard aux emails sur l'agrégation de lien ADSL.

Pour ceux que ça intéresse, je fais partie d'un projet open source qui à la base était un projet de fin d'étude.
Il me semble qu'on a été en contact avec Jérome Nicolle il y a 2 ans je crois, au tout début du projet.

Le projet fonctionne mais à été peu testé. Aucun des devs n'ayant d'infra de test physique, nos tests sont limités.
Le projet est en 'pause' en terme de dev, mais si des gens veulent l'utiliser on est près à s'y remettre pour le faire avancer.

Pour faire une présentation rapide, le système s'installe sur 2 serveurs, un à chaque bout de l'agrégat.
Il tourne en userland, possède une détection de panne et un système de pondération statique.

Aujourd'hui il ne fonctionne que sur Linux, mais le port sous BSD ne doit être très long.

Le lien vers le projet http://agrego.googlecode.com/

Les que l'on avait fait était sur l'agrégation de 2 freebox, avec de l'agrégation sur une dédibox. Le débit atteint était la somme de celles des lignes ADSL moins 10%.

Si ça vous intéresse n'hésitez pas à poser des question sur la mailing list (agrego@googlegroups.com) en français ou en anglais.

Il semblerait que l'utilisation du marking random sur IPTables serait
plus efficace, je vais tenter de me renseigner pour voir si une
intégration pourrait se faire.

Vincent Daligault.

Excerpts from Michel Roche's message of Sat Jul 16 19:03:54 +0200 2011:

> Bonjour,
> je complète juste le message de Jérôme (qui a tout de même vachement
> bien compris le montage, hein ;-), apporte quelques précisions à
> l'attention de Fernando Alves et ajoute les questions correspondantes à
> l'attention de FDN...
>
> Le 12/07/2011 14:33, Jérôme Nicolle a écrit :
>
> > Le MLPPP est un protocole normalisé qui fait sensiblement la même
> > chose avec un tunnel de niveau 3 (PPP), en agrégeant les extrémitées
> > de plusieurs tunnels L2TP. L'inconvénient est qu'il faut alors
> > maitriser le LNS coté opérateur et que ça ne marche que sur des liens
> > PPP à la base (PPPoA ou PPPoE dans le cas de lignes ADSL)
>
> Ça, d'après Benjamin Camat FDN s'en rapproche de plus en plus...
>
> Ce que je ne sais pas du tout par contre, c'est comparer cette solution
> avec celle décrite en dessous en termes de :
> - robustesse : comment gère-t-on les liens qui tombent, dont le débit
> réel disponible varie
> - efficacité : qui des deux est le plus efficace théoriquement,
> peut-être est-ce match nul ?
> - est-il possible d'intégrer une SDSL dans le pool des connexions
> disponibles ?
>
> > L'approche proposée par IELO et dont parle Michel est sensiblement
> > différente. Si j'ai bien compris l'explication que Michel en a faite
> > il y a quelques mois, ça consiste à créer un tunnel de niveau 2 ou 3
> > par lien WAN à destination d'un serveur de terminaison (VM ou dédié),
> > et à créer une table de routage par lien tunnelé. Peu import le type
> > de tunnel d'ailleurs, OpenVPN ou VTun se comporteront de la même
> > manière en l'occurrence.
>
> Oui, nous avons d'ailleurs trois types de liens/tunnels différents chez
> nous :
> - SDSL livré sur un tap
> - ADSL collecté par Ielo avec lien PPPoE monté directement sur la
> machine de collecte
> - ADSL Agrume utilisé grâce à un tunnel OpenVPN monté à l'intérieur de
> celui-ci et aboutissant sur la machine de collecte.
>
> > Des deux cotés (routeur chez toi et serveur de terminaison), tu vas
> > donc marquer les paquets devant passer de l'un à l'autre avec un
> > marqueur aléatoire (weighted random), ou le poids est défini au
> > prorata de la capacité du lien. Tu peux faire des exceptions pour par
> > exemple favoriser un protocole donné sur un lien à faible latence (DNS
> > et SSH sur la ligne SDSL par exemple). Ensuite, tu dis à IPTables de
> > balancer le paquet sur telle ou telle table de routage en fonction u
> > marquage, chaque table définissant comme next-hop l'extrémité opposée
> > d'un des trois tunnels.

> Comme tu peux appliquer des poids différents au marquage de chaque coté des tunnels, ça te permet de gérer le cas d'agrégation de liens symétriques et asymétriques, et tu gardes toute latitude pour moduler le routage de certains paquets en particulier.

Description parfaite : j'en rajoute pas ;-)

> Pour la fiabilité de ce système maintenant, il te faut monitorer les tunnels établis pour que, dès que l'un tombe, tu sois en mesure de recalculer les pondérations sur les autres liens. A ma connaissance personne n'a encore publié le code qui fait ça. Si tu couples ça à un système de mesure de débit (typiquement iperf), tu vas même pouvoir calculer complètement automatiquement la capacité et donc la pondération des liens, et éventuellement faire un test ponctuel qui consiste à considérer un lien comme mort (pour le sortir temporairement de l'agrégat), le benchmarker, puis le réintégrer en ajustant la pondération aux nouvelles valeurs de débit mesuré. Ca peut être utile pour des liens à géométrie variable comme une ligne numéricable ou un lien radio aux heures de pointe.

Ça c'est la partie qui reste à développer. Pour l'instant on a codé un truc bête pour éviter d'utiliser un lien ADSL surchargé à certaines heures bien identifiées, du coup ça passe dans un cron. Mais cette approche n'est pas dynamique du tout. Elle présente seulement l'intérêt d'avoir été très facile à penser et de fonctionner correctement pour nous.

Là il y a du boulot à faire pour améliorer le système : d'une part pour le rendre plus robuste en cas de perte d'un lien, d'autre part pour être plus adaptatif en cas de variation de débit dans les liens. Cette dernière étape n'est pas du tout évidente car il faut parvenir à mettre en place des tests adaptés, suffisamment fréquents pour une bonne réactivité et néanmoins suffisamment discrets pour ne pas perturber le trafic légitime.

> Le seul inconvénient de cette méthode est la diminution du MTU due à l'encapsulation des tunnels sur chaque lien. On pourrait envisager un mécanisme de compression *du payload* (et pas seulement des entêtes) pour réduire la portée de ce problème, à moins biensur que tes applications puissent se contenter d'un MTU de 1320 à 1380 sans que ça grève trop les performances.

Je ne suis pas certain que ce soit réellement un problème. Lorsqu'on m'avait présenté la solution, c'est la première réflexion qui m'était venue à l'esprit. Puis en fait, même si on perd un peu à encapsuler n fois nos paquets, qu'on crée un peu de fragmentation, l'amélioration est telle derrière que franchement c'est insensible. On n'a pas fait changer les MTU des équipements du réseau, ni des abonnés...

> Bref, cette idée a pas mal de potentiel, reste à voir qui aura le temps de la documenter sérieusement et peut être de pondre le bout de code qui manque pour la fiabilisation et la gestion efficace des débits variables ;)

Moi je suis très beaucoup dispo/intéressé pour réfléchir sérieusement à ce problème :-) J'ai même le banc d'essai en live pour effectuer la sélection des tests à effectuer, trouver ceux qui sont pertinents. Après, coder tout ça... chais pas si j'ai les épaules pour.

Pour répondre en partie à la question de Fernando sur Stella. Je ne sais pas ce qu'ils utilisent en vrai puisqu'ils livrent des boites noires/gris foncé aux deux bouts de la connexion. Néanmoins, à voir le détail des docs qu'ils fournissent, je pense que ça ressemble furieusement à ce qu'on a fait. Il y a au moins un RAN (http://lists.vaour.net/mailman/listinfo/ran) qui utilise leurs services avec bonheur (je ne me souviens plus de qui).

La question à FDN maintenant : j'ai bien pris note des développements en cours sur le MLPPP, mais est-ce que ça ne vaudrait pas le coup de proposer et tester également l'autre solution ? Je n'ai pas de vision de leur manière de collecter les ADSL, mais il me semble qu'il serait assez simple de monter une machine virtuelle à mettre à dispo d'un FAI

collecté par FDN et de lui apporter ici ses bouts de ligne. La seule grosse contrainte si je me souviens bien étant que les lignes soient collectées par un free/open BSD seul à même de continuer les tunnels voulus jusqu'à la machine de collecte du FAI (apparemment sous Linux c'est no-way). Il me semble que ça ne ferait pas un trop gros boulot à faire... si l'infra FDN le permet bien sûr. Et pourrait donner à Fernando une réponse immédiate à son problème. Ensuite ça ne fait pas plus de boulot à FDN, le FAI peut faire joujou tout seul de son côté (ou de ses deux côtés de la collecte d'ailleurs :-)
Et là-dessus, même si je ne vaux pas tripette à côté de gradator qui a monté le machin, je veux bien apporter ma contribution, bien volontiers. Comme ça après on sera au moins deux à vouloir bosser sur la fiabilisation du système :-))

Michel

--- End forwarded message ---

#4 - 16/09/2011 16:54 - Laurent GUERBY

http://en.wikipedia.org/wiki/Link_aggregation

Pad planning:

http://piratepad.net/MnP8KdCOXU

#5 - 16/09/2011 17:26 - Laurent GUERBY

Sujet du projet :

Un opérateur utilisant des technologies radio dans son réseau va devoir collecter le trafic de et vers Internet sur un ou plusieurs sites utilisant des technologies filaires comme ADSL, SDSL ou fibre optique. Sur un site de collecte du réseau, pour des raisons de coûts, plusieurs lignes ADSL peuvent être ouvertes en utilisant les multiples paires de cuivres présentes dans tous les logements en France, mais ces lignes peuvent avoir des caractéristiques de débit et latences différentes (venant de l'historique des travaux, ou plus simplement car chaque ligne sera ouverte sur un opérateur différent).

Du coté radio les équipements actuels offrent des débits très largement supérieurs a l'ADSL, de l'ordre de 100 Mbit/s pour un lien de qualité jusqu'à quelques kilomètres. L'objet du projet est de mettre au point une ou plusieurs solutions pour faciliter l'exploitation d'un tel réseau radio collecté sur un ou plusieurs sites en filaire en permettant l'agrégation de lignes en vue d'augmenter le débit disponible et la redondance de site en vue d'augmenter la fiabilité du réseau.

Le cadre théorique général est celui du routage multipath (utilisation de plusieurs métriques afin de déterminer le meilleur chemin), mais ce projet vise une solution spécifique pour des cas de topologie relativement simples mais présents dans la pratique, et non une solution générique.

#6 - 18/09/2011 23:32 - Laurent GUERBY

Fernando ALVES:

après une rencontre avec Laurent Gerby, j'ai moi même commencé l'étude de ce type d'outils (au cas ou mlppp ne fonctionnerait pas).

J'ai fait une petite lecture rapide de agrego:
- Je n'ai pas vu de fragmentation des paquets.
- Dans agrego il faut le même nombres d'interfaces de chaque coté donc pour une agrégation de 4 liens ADSL il faut 8 IP publiques.
- Détection des pannes par PING.

ce que j'ai prévu de faire dans mon outil:
- Fragmentation des paquets ce qui améliore le temps de latence pour les gros paquets.
- Coté serveur une seule interface nécessaire pour la connexion des connexions clientes à agréger.
- Détection des liens en pannes par détection de fragments non reçus (maintien d'un taux d'erreurs par lien), détection rapide d'un lien en erreur et élimination de celui-ci de l'agrégation.
- Possibilité d'ajouter ou enlever à la volé un lien de l'agrégation.
- Maintien d'un taux d'utilisation de chaque liens pour équilibrage de la charge.
- ...

Il me reste à définir le protocole (header des fragments) avant de me lancer.
S'il y a des personnes intéressé par le projet elles sont les bienvenues.

#7 - 22/09/2011 11:34 - Laurent GUERBY

http://chiliproject.tetaneutral.net/projects/tetaneutral/wiki/Projet_agregation
http://lists.tetaneutral.net/listinfo/projet-agregation

#8 - 10/08/2018 09:15 - Matthieu Herrb

- Statut changé de Nouveau à Fermé

fermeture de tous les vieux tickets non suivis depuis plusieurs années