RFC 6301 : A Survey of Mobility Support In the Internet

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 10 juillet 2011. Dernière mise à jour le 12 juillet 2011

Date de publication du RFC : Juillet 2011

https://www.bortzmeyer.org/6301.html

Dans le monde des réseaux informatiques, on distingue en général le **nomadisme** (pouvoir changer d'endroit et avoir à peu près les mêmes services) et la **mobilité** (pouvoir changer d'endroit en maintenant les sessions existantes, comme si on n'avait pas bougé). Aujourd'hui, où beaucoup d'équipements sont mobiles ("*smartphone*", tablette, ...), la mobilité suscite beaucoup d'intérêt. D'où ce RFC qui évalue l'état actuel du déploiement des techniques de mobilité sur l'Internet. (Disons-le tout de suite, il est très faible.)

Je vais me permettre de commencer par une polémique et des opinions personnelles : dans la famille de protocoles TCP/IP, la mobilité, c'est comme le "*multicast*". Beaucoup de RFC, très peu de paquets. Le sujet passionne les experts, pose plein de problèmes techniques amusants, permet d'écrire des algorithmes rigolos... mais touche très peu les utilisateurs. Comment cela, les utilisateurs ne veulent pas se connecter à l'Internet depuis leur maison, puis continuer au bureau ? Si, ils le veulent, mais il n'est pas du tout évident que la mobilité au niveau IP soit nécessaire pour cela. Aujourd'hui, la technique de mobilité la plus courante est DHCP... La machine reçoit une nouvelle adresse IP et ce sont les applications qui gèrent les déconnexions/reconnexions. Prenons l'exemple du client de messagerie instantanée Pidgin et supposons que je me promène avec mon ordinateur portable, connecté via clé USB 3G dans le train, via le Wifi au Starbucks, puis via un câble Ethernet au bureau. J'aurai une adresse IP différente à chaque fois et les sessions IRC ou XMPP seront donc coupées lors des changements. Est-ce grave ? Pas tellement. Pidgin se reconnectera automatiquement à chaque fois et la seule conséquence pratique sera les messages de déconnexion et de reconnexion que verront les abonnés de chaque canal IRC / pièce XMPP. On a là un bon exemple du fait qu'une gestion de la mobilité par l'application cliente (section 5.5 de notre RFC) est suffisante et qu'il n'y a pas besoin d'un service réseau (compliqué et amenant des problèmes de sécurité) pour cela. Même chose avec une application Web : chaque requête HTTP peut utiliser une adresse source différente, l'utilisateur ne s'en rendra pas compte (à part éventuellement une obligation de se réauthentifier si le "*cookie*" est lié à l'adresse IP, ce qui se fait parfois pour limiter la réutilisation d'un "*cookie*" volé). Les applications qui reposent sur HTTP (les clients Twitter, par exemple) arrivent également à maintenir une « session » lors de changements d'adresse IP.

Bien sûr, une telle gestion par l'application ne couvre pas tous les cas. Un gros transfert de fichiers avec curl ou wget ne peut pas fonctionner ainsi, puisque ce transfert utilise une seule connexion TCP, donc dépend de l'adresse IP source utilisée au départ (cf. section 6.2). (Avec rsync, et un script qui le relance jusqu'à completion, ça marcherait.) Et, évidemment, les sessions SSH ne peuvent pas être maintenues lorsqu'on change d'adresse. Mais il reste quand même énormément de cas où il n'y a **pas** de vrai problème lié à la mobilité et cela explique largement, à mon avis, pourquoi les techniques de mobilité IP sont si peu déployées. J'arrête là les opinions personnelles et je reviens au RFC.

La section 1 explique que ce RFC est motivé par le fait que la mobilité est depuis longtemps un sujet de recherche actif à l'IETF et que, depuis quelques années, le déploiement des engins mobiles a explosé ("*smartphones*", tablettes, etc). Le besoin est donc nettement plus marqué maintenant que dix ans auparavant, lorsqu'un ordinateur portable était un machin encombrant et lent. Pourtant, les solutions normalisées par l'IETF ont été peu déployées.

Ces solutions utilisent un vocabulaire rappelé en section 2 et dans le RFC 3753 [1]. Notons notamment les termes d'**identificateur** ("*identifier*") qui indique une valeur qui ne change pas lors des déplacements, de **localisateur** ("*locator*"), qui, lui, change lorsqu'on se déplace (en IP classique, c'est le cas de l'adresse IP), de **correspondance** ("*mapping*"), la fonction qui permet de trouver un localisateur en connaissant l'identificateur, de **correspondant** (CN pour "*Correspondent Node*", la machine avec laquelle l'engin mobile communique), etc.

La section 3 expose les principes de base de la gestion de la mobilité. Le CN doit pouvoir, dans l'Internet actuel :

— trouver l'adresse IP actuelle du mobile (solutions dites de type 1),
— ou bien connaître un identificateur du mobile et pouvoir lui envoyer des données en utilisant cet identificateur (solutions de type 2), nettement plus fréquentes.

Un exemple de solution de type 1 est d'utiliser le DNS. Si le mobile met à jour (par exemple par mise à jour dynamique, cf. RFC 2136) le DNS lorsqu'il se déplace, on utilise le nom de domaine comme identificateur, l'adresse IP comme localisateur et le DNS comme correspondance. Le CN peut donc toujours trouver l'adresse IP actuelle par un simple `getaddrinfo()`. La plupart des techniques normalisées par l'IETF sont, elles, de type 2. Dans celles-ci, l'adresse IP sert d'identificateur (permettant de maintenir les sessions TCP, par exemple dans le cas de SSH). Les paquets envoyés par le CN sont alors transmis à cette adresse IP stable, où un équipement doit ensuite les renvoyer au mobile, alors que les paquets émis par le mobile vers le CN voyagent directement. On parle de routage en triangle. Peu satisfaisant du point de vue des performances, il est en revanche excellent pour la protection de la vie privée : le CN, la machine avec laquelle correspond le mobile, ne sait pas où on est et ne sait pas si on bouge. (Notons que le RFC, et c'est plutôt choquant, ne mentionne pas une seule fois ces questions de protection de la vie privée, même pas dans la section 7 sur la sécurité.)

La section 4 présente ensuite les protocoles de mobilité existants. Je ne vais pas les reprendre tous ici (la liste en compte vingt-deux !) mais simplement en choisir arbitrairement quelques uns. On trouve dans cette liste des ancêtres comme le protocole Columbia de 1991, ou des protocoles plus récents comme le Mobile IP de 1996. Certains des protocoles étudiés n'ont pas été conçus uniquement pour la mobilité, mais la facilitent. C'est le cas de HIP <https://www.bortzmeyer.org/hip-resume.html> (2003), ILNP (2005) ou LISP (2009).

Columbia, par exemple, le premier conçu, ne fournissait la mobilité qu'à l'intérieur du campus. Dans chaque cellule radio, un routeur spécial, le MSS ("*Mobile Support Station*"), assurait le routage des paquets dont l'adresse IP source appartenait au préfixe spécial « mobilité ». Le mobile gardait donc son adresse et les MSS s'arrangeaient entre eux pour acheminer le paquet.

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc3753.txt`

Le principal protocole standard de mobilité est aujourd'hui Mobile IP, normalisé dans les RFC 3344, RFC 6275 et RFC 5454. Il fonctionne en IPv4 ou en IPv6, sur la base des mêmes principes. Chaque mobile a un "*Home Agent*", qui ne bouge pas, et qui fournit au mobile une adresse stable ("*Home Address*"). Le mobile a aussi une adresse IP liée à son point d'attache actuelle, l'adresse CoA ("*Care-of Address*"). Le mobile prévient le "*Home Agent*" de tout changement de CoA. Le correspondant, le CN, écrit toujours à l'adresses "*Home Address*", charge au "*Home Agent*" de faire suivre. Mobile IP est donc de type 2 et, par défaut, fait du routage en triangle (une optimisation permet de le supprimer dans certains cas). Il existe aujourd'hui de nombreuses mises en œuvre de Mobile IP, aussi bien pour les mobiles que pour les "*Home Agents*" mais très peu de déploiements.

Un exemple de protocole de type 1 est E2E. Dans ce cas, l'identificateur stable du mobile est un nom de domaine, et c'est une simple requête DNS qui indique au correspondant quelle adresse IP utiliser. Le mobile, lorsqu'il change d'adresse IP, utilise les mises à jour dynamiques du DNS pour modifier cette information.

HIP n'était pas conçu uniquement pour la mobilité. Toutefois, en séparant identificateur et localisateur <https://www.bortzmeyer.org/separation-identificateur-localisateur.html>, il facilite celle-ci (RFC 5206). Pour trouver une adresse IP à partir d'un HI ("*Host Identifier*", les identificateurs stables des nœuds HIP), on peut utiliser le DNS mais aussi des « serveurs de rendez-vous » spécifiques à HIP (RFC 5204). La mobilité avec HIP nécessite que le CN soit un nœud HIP également. Lorsqu'un mobile HIP se déplace, il doit prévenir à la fois les CN et le serveur de rendez-vous (message HIP `UPDATE`, section 5.3.5 du RFC 5201).

Autre protocole de séparation de l'identificateur et du localisateur qui facilite la mobilité, LISP. Bien plus récent, il est encore peu déployé, et ses extensions de mobilité sont seulement sous forme d'un projet de RFC, `draft-meyer-lisp-mn`.

Armé de ces descriptions, le RFC, dans se section 5, explique les différents choix qui ont été faits pour chaque protocole. C'est la section la plus intéressante du RFC. (La section 6.3 examine plus en détail les avantages et inconvénients de chaque approche.) Par exemple, une approche était de réutiliser le routage, en gardant l'adresse IP fixe pour le mobile, et en comptant sur des protocoles proches de ceux de routage pour assurer l'acheminement du paquet jusqu'au bonne endroit. Le protocole Columbia fonctionnait ainsi. Mais de telles solutions ne marchent bien qu'en réseau local : pas question de changer les routes de l'Internet chaque fois qu'un mobile se déplace.

Deuxième approche : une fonction de correspondance entre un identificateur stable et l'adresse IP actuelle. Au lieu de prévenir le monde entier lorsque son adresse change, le mobile n'aurait plus à prévenir qu'un seul point (le serveur DNS, ou le serveur de rendez-vous, pour les solutions de type 1 ou bien le "*Home Agent*", pour celles de type 2.) Si le CN n'est pas prévenu du changement (cas de Mobile IP, par défaut), on perd en performances (routage en triangle) mais on gagne en intimité et surtout on n'a pas besoin que le CN comprenne le protocole de mobilité (avec Mobile IP, le mobile peut parler à des machines IP normales, qui ignorent tout de ce protocole).

Ce dernier point, couvert en détail dans la section 5.2, est crucial : qui doit être mis au courant de la mobilité ? Il y a quatre parties concernées : le mobile, son correspondant (CN), le réseau (les routeurs) et le composant qui donne un coup de main ("*Home Agent*", serveur DNS, etc). Si le CN doit être mis au courant (cas de HIP, par exemple), il faut mettre à jour toutes les machines avec qui le mobile est susceptible de communiquer, ce qui ne semble pas très réaliste (d'autant plus que les gros serveurs Internet auraient à garder un état considérable, pour se souvenir de tous leurs clients récents). La plupart des approches, à commencer par Mobile IP, gardent donc une adresse IP stable et le CN est une machine ordinaire, qui ne sait même pas qu'elle parle à un mobile. Il existe aussi quelques protocoles où ni le CN, ni le mobile, ne sont au courant et où le réseau fait tout. Dans le futur, toutefois, de plus en plus de

machines seront mobiles et il est possible que le premier choix (rendre le CN conscient que son partenaire se déplace) redevienne attirant.

Autre choix à faire, la mobilité doit-elle être contrôlée par le réseau ou par l'utilisateur (section 5.3) ? Dans les réseaux de téléphonie mobile, le terminal ne gère pas la mobilité, le réseau fait tout pour lui, et ça marche. Mais cela prive l'utilisateur de tout contrôle sur la façon dont est gérée la mobilité. (Songez aux scandaleux tarifs d'itinérance, rendus possibles par le fait que l'utilisateur ne peut pas empêcher que tout passe par son opérateur habituel.)

Les différences entre les protocoles de mobilité s'expliquent aussi par le fait que certains visent une solution mondiale, qui doit marcher pour des centaines de millions de mobiles se déplaçant partout dans le monde (ce qui soulève de sérieux problèmes de "*scalability*") alors que d'autres (comme Columbia, cité plus haut), ne cherchent qu'une mobilité locale, problème bien plus facile (section 5.4).

Enfin, les vingt-deux protocoles examinés dans ce RFC ne couvrent pas tout. On peut aussi baptiser « mobilité » des solutions simples comme un serveur OpenVPN - qui jouera un rôle équivalent à celui du "*Home Agent*" - plus un client OpenVPN sur chacune des machines de l'utilisateur, avec une adresse IP fixe chacune. Il existe aussi des approches radicalement différentes (section 5.5) comme GTP (qui ne marche qu'à l'intérieur d'un fournisseur donné) ou comme des solutions fondées sur les applications, comme je l'ai prôné au début de cet article. Par exemple, SIP dispose d'une extension pour gérer la mobilité (cf. Schulzrinne, H. et E. Wedlund, « "*Application-Layer Mobility Using SIP*" `<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.95.3546&rep=rep1&type=pdf>` », dans "*Mobile Computing and Communications Review*", 2010) où un nouveau message `INVITE` est envoyé lorsque le mobile change d'adresse IP pendant une communication.

Je l'ai dit, les protocoles spécifiques de mobilité sont pour l'instant un échec complet. Mais que prévoient les auteurs du RFC pour le futur ? La section 6 analyse d'abord cet échec (section 6.1). Peu ou pas de déploiement en production, donc. Mais pourquoi ? Le RFC estime que c'est en partie le résultat du fait que les machines mobiles ne sont devenues banales que depuis très peu de temps. Mais il reconnait aussi que les protocoles de mobilité sont des usines à gaz complexes et lentes. Certains suggèrent même de simplifier le problème en ignorant les questions de performance et de sécurité (cf. section 7 sur les problèmes de sécurité spécifiques à la mobilité) dans un premier temps (ce conseil vient directement de l'excellente section 3 du RFC 5218).

Puis la section 6 explore les pistes à suivre. Une très bonne lecture pour les concepteurs de protocoles et les étudiants en réseaux informatiques.

Pour la mobilité des téléphones et la fréquence des déconnexions, changements d'adresse IP, etc, je recommande le bon article « "*A Day in the Life of a Mobile Device*" `<http://urbanairship.com/blog/2011/07/07/a-day-in-the-life-of-a-mobile-device-ip-connectivity/>` ».