

# HTTP/2 : une évolution importante du protocole du web, notamment pour les mobiles | Denis JACOPINI



#HTTP/2 : une évolution importante du protocole du web, notamment pour les mobiles

**Le protocole HTTP a 20 ans. On a récemment parlé de son évolution HTTP/2 avec la possibilité de chiffrer systématiquement les échanges sur le web, mais cela va bien au-delà et pourrait constituer une évolution importante, notamment pour les mobiles.**

Le protocole HTTP, support historique du web avec le langage HTML, a été inventé par Tim Berners-Lee il y a maintenant 20 ans. On a récemment parlé de son évolution HTTP/2, notamment dans le contexte d'un possible chiffrage systématique des échanges entre navigateurs et serveurs, après les affaires d'écoutes illégales des communications Internet révélées par Edward Snowden. Pour autant, cette amélioration à venir de HTTP va bien au-delà et pourrait constituer une étape de progrès importante pour le web, notamment depuis les mobiles. Vous avez peut-être lu récemment des articles sur HTTP/2, dans le contexte d'une volonté de sécurisation accrue des échanges de données sur le web. Mais comme l'explique #Mark Nottingham, responsable du groupe de travail HTTP/2 à l'IETF - #Internet Engineering Task Force - les attentes de la communauté du web vont au-delà, et les évolutions prévues sont plus larges.



Mark Nottingham, responsable du groupe de travail HTTP/2 à l'IETF

Car cela fait maintenant 20 ans que le protocole HTTP - #HyperText Transfer Protocol - a été inventé, et depuis les versions HTTP/1.0 en mai 1996 et #HTTP/1.1 en janvier 1997 il n'a plus évolué. Dans le même temps l'Internet s'est diffusé massivement dans le grand public et en entreprise, et en moins de deux décennies on est passé de connexions téléphoniques à 56 kbit/s à des dizaines de Mbit/s sur des liaisons DSL ou fibre optique, et la part du trafic Internet réalisée par des smartphones, tablettes et terminaux mobiles de toute sortes a explosé. Afin de pallier les limitations de HTTP, Google a mené ses propres travaux, dévoilant #SPDY en 2012. Cette proposition visait essentiellement à réduire le temps de chargement des pages web en priorisant les contenus nécessaires à leur affichage et en multipliant leur transfert au sein d'une seule connexion TCP. Ces travaux ont inspiré l'IETF qui a repris cette approche dans un cadre plus global d'amélioration des performances du web.

Faisons donc le point sur les axes d'évolution du protocole HTTP, en listant les principaux apports techniques de HTTP/2.

#### Conserver les mêmes API

La compatibilité ascendante est primordiale, compte tenu de nombre colossal de services qui ont été développés au-dessus du protocole HTTP : on ne saurait exiger des développeurs de logiciels utilisant HTTP qu'ils retouchent leur code, parfois embarqué dans des systèmes matériels, pour l'adapter. Les mêmes méthodes (GET, PUT, POST etc), les mêmes entêtes et les mêmes statuts et code d'erreur (le fameux 404 et tous les autres) seront conservés inchangés, et il suffira de remplacer une bibliothèque de fonctions HTTP/1.1 par une bibliothèque HTTP/2 pour que le logiciel qui l'utilise soit compatible.

#### Rendre les requêtes moins coûteuses

HTTP est un protocole coûteux en ressources réseau, notamment parce que les entêtes de messages sont verbeux. C'est pénalisant pour l'échange de courtes informations, et ce l'est encore plus pour les mobiles : même si les débits ont été fortement accrues avec la 3G puis la 4G, tout ce qui contribue à optimiser la bande passante des réseaux mobiles est bénéfique. Pour cela, http va utiliser un mécanisme de « multiplexage » qui permettra l'échange de plusieurs messages simultanément, et non plus séquentiellement. De surcroît, les entêtes seront systématiquement compressées.

#### Optimiser les connexions TCP avec les serveurs

Alors qu'aujourd'hui HTTP permet d'ouvrir plusieurs connexions TCP parallèles et simultanées vers un serveur, ce qui peut créer de la congestion sur le réseau, HTTP/2 permettra de regrouper les échanges avec un même serveur au sein d'une seule et même connexion TCP. Là encore, ce sera bénéfique pour les réseaux mobiles.

#### Pré-remplir le cache des navigateurs

Un mécanisme de « cache pushing » permettra à un serveur d'envoyer des informations à un client pour un usage ultérieur. Ainsi il ne sera plus nécessaire à un navigateur web de demander d'abord le contenu d'une page HTML, puis les feuilles de style CSS référencées par la page : le serveur pourra envoyer le HTML et les CSS d'un seul coup, évitant un dialogue inutile entre le navigateur et le serveur. Le logiciel client pourra évidemment débrayer ce mécanisme si nécessaire.

#### Interrompre une connexion TCP sans la fermer

Avec HTTP, si un logiciel client envoie une requête et décide de ne plus attendre la réponse, la seule façon de préserver la bande passante est de fermer la connexion TCP. Il n'y a aucune façon de la récupérer si cela s'avère nécessaire plus tard.

Avec HTTP/2, un logiciel client pourra maintenir active une connexion TCP même s'il abandonne un échange en cours à la suite d'une action de l'utilisateur, ce qui rendra moins coûteuse en ressources réseaux une éventuelle reprise du dialogue ultérieurement.

#### Faciliter le chiffrage des échanges

C'est là le point qui a le plus attiré l'attention sur HTTP/2, et qui n'a peut-être pas été bien compris : la nouvelle version du protocole ne rendra pas obligatoire l'utilisation du chiffrage #TLS, sur lequel repose le chiffrage #SSL entre sites web et navigateurs. En revanche, ses performances accrues réduiront l'impact négatif du chiffrage sur la rapidité de réponse d'un site telle que la perçoivent les utilisateurs.

En rendant le chiffrage des communications plus performant, HTTP/2 pourra contribuer à rendre le web plus sûr pour ses utilisateurs, tant vis-à-vis de la cybercriminalité que d'états qui, on le voit aujourd'hui, n'hésitent pas à espionner massivement les internautes de toutes nationalités. Pour autant, au regard des moyens technologiques colossaux dont s'est dotée la NSA, il ne faudrait pas croire en une sécurité absolue d'un tel chiffrage.

#### Pour autant, HTTP/2 n'est pas magique

On l'a vu, HTTP/2 apporte de nombreuses évolutions positives. Mais il n'aura pas d'impact magique sur les sites web : afin d'en tirer le meilleur, les évolutions des navigateurs web, et aussi des logiciels serveurs web, devront être faites en prenant en compte différents scénarios d'usage. En effet, s'ils sont mal utilisés, des mécanismes tels que le pré-remplissage du cache du navigateur ou le maintien de sessions TCP inactives pourraient aller à l'encontre du but recherché : l'amélioration globale de la performance du web.

#### Et ensuite, HTTP/3 ?

HTTP existe depuis 20 ans : sa simplicité et sa versatilité ont permis l'incroyable développement du web que l'on connaît aujourd'hui. Le développement de HTTP/2 a été très compliqué, notamment parce qu'un grand nombre de matériels ont été conçus dans l'idée que HTTP ne changerait jamais. Une fois la transition de HTTP à HTTP/2 lancée, les choses seront différentes, et de futures nouvelles versions pourront être introduites plus facilement.

Parmi des évolutions déjà envisagées, certaines n'ont pas été intégrées dans HTTP/2 pour ne pas retarder encore sa sortie. Sont notamment prévus l'envoi au client de certificats TLS et d'entrées DNS. Des problèmes non encore résolus par HTTP/2 pourront l'être dans une version future, mais cela devrait être marginal, car la communauté est confiante dans HTTP/2, au vu des premiers tests de déploiement déjà réalisés. L'objectif de tous est maintenant qu'il sorte rapidement et commence à être déployé.

Les résultats des travaux de l'IETF sur les spécifications techniques de HTTP/2 sont publics. Le draft 10 (<http://tools.ietf.org/html/draft-ietf-httpbis-http2-10>) peut être consulté et va être discuté durant une période de 6 mois prenant fin en août 2014.

Après cette lecture, quel est votre avis ?

(Cliquez et laissez-nous un commentaire.)

Source : <http://www.zdnet.fr/actualites/http-2-une-evolution-importante-du-protocole-du-web-notamment-pour-les-mobiles-39798198.htm>  
par Pierre Col