Groupe de travail Réseau
Request for Comments : 1958
Catégorie : Information
Traduction Claude Brière de l'Isle

B. Carpenter, éditeur
IAB
juin 1996

Principes de l'architecture de l'Internet

Statut de ce mémoire
Le présent mémoire donne des informations pour la communauté de l'Internet. Il ne spécifie aucune norme de l'Internet. Sa distribution n'est soumise à aucune restriction.

Résumé
L'Internet et son architecture ont crû de façon évolutive à partir de débuts modestes et non pas à partir d'un grand plan. Alors que ce processus d'évolution est une des principales raisons du succès de cette technologie, il semble néanmoins utile de faire un point sur les principes actuels de l'architecture de l'Internet. Ce document est destiné à donner des lignes directrices globales dans l'intérêt général, et en aucune façon à être un modèle de référence formel ou invariant.

Table des Matières

1. Un changement constant

En recherchant les principes architecturaux de l'Internet, on doit se souvenir que le changement technique est continu dans l'industrie des technologies de l'information. L'Internet en est le reflet. Depuis les 25 années écoulées depuis le début de l'ARPANET, diverses mesures de la taille de l'Internet ont montré des accroissements d'un ordre de grandeur compris entre 1000 (vitesse du cœur de réseau) et 1 000 000 (nombre d'hôtes). Dans cet environnement, certains principes architecturaux changent inévitablement. Des principes qui semblaient inviolables quelques années auparavant sont déconseillés aujourd'hui. Des principes qui semblent sacrés aujourd'hui seront déconseillés demain. Le principe d'un changement constant est peut-être le seul principe de l'Internet qui pourrait survivre indéfiniment.

L'objet du présent document n'est donc pas de graver dans le marbre des dogmes sur la façon dont les protocoles de l'Internet devraient être conçus, ou même sur la façon dont ils devraient s'harmoniser. C'est plutôt d'apporter diverses lignes directrices qui ont été trouvées utiles dans le passé, et qui pourraient être utiles à ceux qui conçoivent de nouveaux protocoles ou qui évaluent de tels concepts.

Une bonne analogie pour le développement de l'Internet est celle du renouvellement constant des rues et bâtiments individuels d'une ville, plutôt que de raser la ville et de la reconstruire. Les principes architecturaux visent donc à fournir un cadre pour créer une coopération et des normes, comme une petite "famille de générateurs" de règles qui génèrent un grand espace de technologies variées et évolutives.

Parmi les déclencheurs techniques actuels du changement figurent les limites de la capacité d'échelonnement de IPv4, le fait que les réseaux en gigabit/s et le multimédia présentent fondamentalement de nouveaux défis, ainsi que le besoin de garanties de qualité de service et de sécurité dans l'Internet commercial.

Comme le déclarait Lord Kelvin en 1895, "Les appareil volants plus lourds que l'air sont impossibles." Il serait

déraisonnable d'imaginer que les principes dont la liste figure ci-dessous sont plus qu'un cliché de notre compréhension actuelle.

2. Y a-t-il une architecture de l'Internet ?

2.1 De nombreux membres de la communauté Internet vont dire qu'il n'y a pas d'architecture, mais seulement une tradition, qui n'a pas été couchée par écrit pendant les 25 premières années (ou tout au moins, pas par l'IAB). Cependant, en termes très généraux, la communauté estime que le but est la connexité, que l'outil est le protocole Internet, et que l'intelligence est de bout en bout plutôt que cachée dans le réseau.

La croissance exponentielle actuelle du réseau semble montrer que la connexité est sa propre récompense, et a plus de valeur que n'importe quelle application individuelle telle que la messagerie ou la Toile mondiale. Cette connexité exige la coopération technique entre les fournisseurs de service, et elle fleurit dans l'environnement de télécommunications commerciales de plus en plus libérales et concurrentielles.

La clé de la connexité globale est la couche inter-réseaux. La clé de l'exploitation de cette couche à travers des divers matériels qui produisent la connexité globale est "l'argument de bout en bout".

2.2 On pense généralement que dans une situation idéale, il ne devrait y avoir qu'un seul protocole au niveau de l'Internet. Cela permet des opérations uniformes et relativement transparentes dans un réseau en concurrence avec plusieurs fabricants et plusieurs fournisseurs de services. Il peut, bien sûr y avoir plusieurs protocoles pour satisfaire des exigences différentes à d'autres niveaux, et il y a de nombreux exemples réussis de grands réseaux privés qui utilisent plusieurs protocoles de couche réseau.

En pratique, il y a au moins deux raisons pour lesquelles un seul protocole de couche réseau devrait être utilisé sur l'Internet public. D'abord, il peut y avoir le besoin d'une transition graduelle d'une version de IP à une autre. Ensuite, des exigences fondamentalement nouvelles peuvent conduire à un protocole fondamentalement nouveau.

Le protocole de niveau Internet doit être indépendant du support matériel et de l'adressage matériel. Cette approche permet à l'Internet d'exploiter toute nouvelle technologie de transmission numérique de quelque sorte qu'elle soit, et de découpler ses mécanismes d'adressage du matériel. Cela permet à l'Internet d'être facile à interconnecter sur des supports de transmission fondamentalement différents, et d'offrir une plate-forme unique pour une grande variété d'applications et de services d'infrastructure d'informations. Il y a une bonne présentation de ce modèle et d'autres questions d'importance fondamentale dans [Clark].

2.3 On pense généralement aussi que les fonctions de bout en bout sont mieux réalisées par des protocoles de bout en bout.

L'argument de bout en bout est exposé en détail dans [Saltzer]. L'argument de base est que, comme premier principe, certaines fonctions de bout en bout exigées ne peuvent être effectuées correctement que par les systèmes d'extrémité eux-mêmes. Un cas spécifique est que tout réseau, quel que soit le soin de sa conception, sera sujet à des défaillances de transmission à un taux statistiquement déterminé. La meilleure façon de s'en accommoder est de l'accepter, et de donner la responsabilité de l'intégrité de la communication aux systèmes d'extrémité. Un autres cas spécifique est la sécurité de bout en bout.

Pour citer [Saltzer], "La fonction en question ne peut être mise en œuvre complètement et correctement qu'avec la connaissance et l'aide de l'application qui se tient aux points d'extrémité du système de communications. Donc, fournir la fonction en question comme caractéristique du système de communication lui-même n'est pas possible. (Parfois une version incomplète de la fonction fournie par le système de communication peut être utile au titre de l'amélioration des performances.")

Ce principe a des conséquences importantes si on exige des applications qu'elles survivent à des défaillances partielles du réseau. Une conception de protocole de bout en bout ne devrait pas s'appuyer sur la maintenance d'état (c'est-à-dire, des informations sur l'état de la communication de bout en bout) à l'intérieur du réseau. Un tel état devrait n'être maintenu que dans les points d'extrémité, d'une façon telle que l'état ne puisse être détruit que lorsque le point d'extrémité a lui-même une défaillance (appelée partage de sort *(fate-sharing)*). Une conséquence immédiate en est que les datagrammes sont mieux que des circuits virtuels classiques. Le travail du réseau est de transmettre les datagrammes aussi efficacement et souplement que possible.

Tout le reste devrait être effectué sur les extrémités.

Pour rendre ses services, le réseau maintient certaines informations d'état : routes, garanties de qualité de service qu'il assure, informations de session lorsque elles sont utilisées dans la compression d'en-tête, historique de compression pour la compression des données, et ainsi de suite. Cet état doit être auto réparant ; des procédures ou protocoles adaptifs doivent exister pour déduire et maintenir cet état, et le changer quand la topologie ou l'activité du réseau change. Le volume de cet état doit être minimisé, et la perte de l'état ne doit pas résulter en plus qu'un déni de service temporaire étant donné que cette connexité existe. Les états configurés manuellement doivent être gardé à un minimum absolu.

2.4 Heureusement l'Internet n'appartient à personne, il n'y a pas de contrôle centralisé, et personne ne peut le débrancher. Son évolution dépend d'un vague consensus sur des propositions techniques, et sur un code de fonctionnement. Les retours sur l'ingénierie des mises en œuvre réelles sont plus importants que tous les principes architecturaux.

3. Questions de conception générale

3.1 L'hétérogénéité est inévitable et doit être prise en charge par conception. Plusieurs types de matériels doivent être permis, par exemple, des vitesses de transmission différant d'au moins 7 ordres de grandeur *(10^7 ?)* , des longueurs de mots d'ordinateurs différentes, et des hôtes allant du micro ordinateur aux faibles ressources de mémoire au super calculateur massivement parallèle. Plusieurs types de protocoles d'application doivent être permis, allant du plus simple, comme celui à connexion à distance jusqu'au plus complexe comme des bases de données réparties.

3.2 Si il y a plusieurs façon de faire la même chose, il faut en choisir une. Si une conception antérieure, dans le contexte de l'Internet ou ailleurs, a résolu avec succès le même problème, choisir cette même solution sauf s'il existe une bonne raison technique de ne pas le faire. La duplication des mêmes fonctionnalités de protocole devrait être évitée autant que possible, sans bien sûr utiliser cet argument pour rejeter les améliorations.

3.3 Tous les concepts doivent être prêts à s'adapter à de très nombreux nœuds par site et à de nombreux millions de sites.

3.4 Les performances et le coût doivent être pris en compte autant que les fonctionnalités.

3.5 Faire simple. Lorsque vous avez un doute au moment de la conception, choisissez la solution la plus simple.

3.6 La modularité est bonne. Si les choses peuvent rester séparées, qu'elles le soient.

3.7 Dans de nombreux cas, il est préférable d'adopter maintenant une solution presque complète, plutôt que d'attendre qu'une solution parfaite soit trouvée.

3.8 Éviter les options et les paramètres chaque fois que possible. Toutes les options et tous les paramètres devraient être configurés ou négociés de façon dynamique plutôt que manuellement.

3.9 Être strict à l'envoi et tolérant à la réception.
Les mises en œuvre doivent suivre précisément les spécifications lorsqu'elles envoient au réseau, et tolérer des entrées fautives provenant du réseau. Dans le doute, éliminer en silence les entrées fautives, sans retourner de message d'erreur à moins que ce ne soit exigé par la spécification.

3.10 Être parcimonieux dans l'envoi de paquets non sollicités, en particulier de diffusion et de diffusion groupée.

3.11 Les interdépendances circulaires doivent être évitées.
Par exemple, l'acheminement ne doit pas dépendre de recherches dans le système des noms de domaines (DNS, *Domain Name System*), car la mise à jour des serveurs du DNS dépend de la réussite de l'acheminement.

3.12 Les objets devraient être auto-descriptifs (inclure le type et la taille), dans des limites raisonnables. Seuls les codes de type et autres numéros magiques alloués par l'Autorité d'allocation des numéros de l'Internet (IANA, *Internet Assigned Numbers Authority*) peuvent être utilisés.

3.13 Toutes les spécifications devraient utiliser la même terminologie et la même notation, et la même convention de l'ordre des bits et des octets.

3.14 Et, peut-être plus important : Rien ne peut être normalisé tant qu'il n'y a pas plusieurs instances de code en fonction.

4. Questions de nom et d'adresse

4.1 Éviter tout concept qui exige que les adresses soient gravées dans le matériel ou mémorisées dans un support non volatile (excepté bien sûr lorsque c'est une exigence essentielle comme dans un serveur de noms ou de configuration). En général, les applications d'utilisateur devraient utiliser des noms plutôt que des adresses.

4.2 Une seule structure de dénomination devrait être utilisée.

4.3 Les noms publics (c'est-à-dire, largement visibles) devraient être en ASCII indépendant de la casse. Cela se réfère précisément aux noms du DNS, et aux éléments de protocole qui sont transmis en format texte.

4.4 Les adresses doivent être sans ambiguïté (uniques au sein de toute leur portée possible).

4.5 Les protocoles de couche supérieure doivent être capables d'identifier sans ambiguïté les points d'extrémité. En pratique aujourd'hui, cela signifie que les adresses doivent être les mêmes au début et à la fin de la transmission.

5. Questions externes

5.1 Donnez la préférence aux technologies non brevetées, mais si la meilleure technologie est brevetée et est disponible à tous à des conditions raisonnables, l'incorporation d'une technologie couverte par un brevet est acceptable.

5.2 L'existence de contrôle à l'export sur certains aspects de la technologie de l'Internet est seulement d'importance secondaire dans le choix de la technologie à adopter dans les normes. Toute la technologie exigée pour mettre en œuvre les normes de l'Internet peut être fabriquée dans n'importe quel pays, aussi le développement mondial de la technologie de l'Internet ne dépend pas de la possibilité d'exporter vers un pays ou groupe de pays particulier.

5.3 Aucune mise en œuvre qui ne comporte pas tous les composants exigés ne peut revendiquer la conformité à la norme.

5.4 Les concepts devraient être pleinement internationaux, avec la prise en charge de la localisation (adaptation aux jeux de caractères locaux). En particulier, il devrait y avoir une approche uniforme de l'étiquetage des jeux de caractères à des fins d'information.

6. Questions de confidentialité et d'authentification

6.1 Tous les concepts doivent trouver leur place dans l'architecture de sécurité d'IP.

6.2 Il est très souhaitable que les transporteurs de l'Internet protègent la confidentialité et l'authenticité de tout trafic, mais ceci n'est pas une exigence de l'architecture. La confidentialité et l'authentification sont de la responsabilité des utilisateurs finaux et elles doivent être mises en œuvre dans les protocoles utilisés par les utilisateurs finaux. Les points d'extrémité ne devraient pas dépendre de la confidentialité ou de l'intégrité assurée par les transporteurs. Les transporteurs peuvent choisir de fournir un certain niveau de protection, mais ceci est accessoire à la principale responsabilité des utilisateurs finaux de se protéger eux-mêmes.

6.3 Chaque fois qu'un algorithme cryptographique est invoqué dans un protocole, le protocole devrait être conçu pour permettre d'utiliser des algorithmes de remplacement et l'algorithme spécifique utilisé dans une mise en œuvre particulière devrait être explicitement étiqueté. Les étiquettes officielles pour les algorithmes sont à enregistrer auprès de l'IANA.
(On peut discuter de la généralisation de ce principe au delà de la sphère de la sécurité.)

6.4 Dans le choix d'un algorithme, ce doit être celui qui est largement considéré comme assez fort pour servir l'objet auquel il est destiné. Parmi ceux qui sont de force suffisante et égale, la préférence devrait être donnée aux algorithmes qui ont subi avec succès l'épreuve du temps, et qui ne sont pas inefficaces sans nécessité .

6.5 Pour assurer l'interopération entre des points d'extrémité qui utilisent des services de sécurité, un algorithme (ou une suite d'algorithmes) devrait être rendue obligatoire pour assurer la capacité à négocier un contexte sûr entre les mises en œuvre. Sans cela, les mises en œuvre risqueraient de ne pas avoir d'algorithme en commun et de n'être pas capables de communiquer en toute sécurité.

Remerciements

Le présent document est un travail collectif de la communauté de l'Internet, publié par le Bureau de l'architecture de l'Internet (IAB). Des remerciements particuliers à Fred Baker, Noel Chiappa, Donald Eastlake, Frank Kastenholz, Neal McBurnett, Masataka Ohta, Jeff Schiller et Lansing Sloan.

Références

Noter que les références ont été délibérément limitées aux deux articles fondamentaux sur l'architecture de l'Internet.

[Clark] The Design Philosophy of the DARPA Internet Protocols, D.D.Clark, Proc SIGCOMM 88, ACM CCR Vol 18, n° 4, août 1988, pages 106-114 (réimprimé dans ACM CCR Vol 25, n° 1, janvier 1995, pages 102-111).

[Saltzer] End-To-End Arguments in System Design, J.H. Saltzer, D.P.Reed, D.D.Clark, ACM TOCS, Vol 2, n° 4, novembre 1984, pp 277-288.

Considérations pour la sécurité

Les questions de sécurité sont exposées tout au long du présent mémoire.

Adresse de l'éditeur

Brian E. Carpenter
Group Leader, Communications Systems
Computing and Networks Division
CERN
European Laboratory for Particle Physics
1211 Genève 23,
Confédération Helvétique

téléphone : +41 22 767-4967
fax : +41 22 767-7155
mél : brian@dxcoms.cern.ch