COPENHAGUE – Comment ça marche : Les bases du DNS
Dimanche 12 mars 2017 – 11 h à 12 h 30 CET
ICANN58 | Copenhague, Danemark

STEVE CONTE: … Si vous êtes des connaisseurs du DNS, je pense que vous allez vous moquer de moi. Donc, je ne vous conseille pas de rester. Nous allons commencer. Donc, merci de nous avoir rejoint aujourd'hui. Je suis Steve Conte. Je suis directeur du programme du bureau du CTO de l'ICANN. Nous voilà pour une leçon de base sur le DNS. Si vous opérez des serveurs de DNS, vous vous êtes trompé de salle. Si vous êtes là pour rattraper et avoir de nouvelles informations sur le DNS, vous êtes dans la bonne salle.

Nous essayons de tenir de temps à autre ce type de session et entre aujourd'hui et demain, nous allons avoir ces sessions d'initiation sur les différentes parties de la technologie qui apparaissent dans la sphère de l'Internet ou du système Internet de l'ICANN. Et cette semaine, nous essayons d'organiser ce type de session d'initiation au DNS où vous êtes.

On a également d'autres sessions sur le réseautage sur Internet. On a Alain Durand du bureau du CTO ; de l'équipe d'APNIC, on a

quelqu'un d'autre qui va venir présenter d'autres informations aussi.

Nous avons l'abus du DNS et une présentation pour comprendre les cas d'abus du DNS, et le bureau de John Crain de l'ICANN qui va nous expliquer comment utiliser le DNS et quels seraient les cas d'abus liés au DNS pour nous permettre de mieux comprendre comment atténuer ces cas d'abus.

Et finalement, nous avons le comité consultatif du système de serveur racine, le RSSAC, qui présentera cette après-midi ce que cela signifie que d'être un opérateur du serveur racine, ce qu'est un serveur racine et le rôle des opérateurs.

C'est une session qui est toujours très intéressante et je vous invite à y participer. Si vous n'êtes pas en mesure de nous rejoindre aujourd'hui, mais que vous êtes intéressé par ces questions, nous aurons de nouvelles sessions à ces sujets demain. Si vous regardez le programme de demain et que vous voyez les mêmes sessions sur comment tout fonctionne, ce sont les mêmes sessions que nous allons présenter aujourd'hui. Mais si vous n'êtes pas disponible aujourd'hui, donc je vous recommande d'essayer de nous rejoindre demain. Cela dit, demain, nous allons entrer directement, maintenant, dans les aspects fondamentaux du DNS.

Le principal à savoir est que les numéros sont difficiles à se rappeler. Il est facile de se rappeler de numéros de téléphone, mais s'il vous fallait apprendre tout votre agenda et tous les numéros de téléphone que vous avez c'est difficile, et c'est pareil pour les adresses IP. On a des milliards d'adresses IP sur Internet. En IPv4, dans l'espace IPv4, il y a des milliards d'adresses.

Il y a encore plus dans l'espace IPv6 et ça devient de plus en plus difficile de se rappeler des adresses IP de serveur spécifique. Donc, le système des noms de domaine, c'est-à-dire le DNS a été développé pour pouvoir associer ou mettre en rapport des mots avec des adresses IP sur Internet. Les noms sont beaucoup plus faciles à se rappeler que les numéros.

Au départ, les noms étaient tous simples, il n'y avait que quelques serveurs sur Internet à l'époque. Donc, il était facile de se rappeler de ses noms d'adresse qui n'avaient qu'une étiquette. Il n'y avait pas de point, rien à la fin. Ce n'était qu'un nom de page et c'était tout. Ça faisait référence à un nom d'hébergement. Le concept de repérage pour mettre en rapport un nom avec une adresse IP s'appelle le système de résolution de nom.

La résolution de nom au départ était hébergée dans un fichier mère qui était gardé dans un ordinateur, qui était enregistré

dans un ordinateur. Et tous les ordinateurs qui étaient connectés à Internet accédaient à ce fichier qui avait un répertoire de noms liés à des adresses IP qui étaient hébergées dans d'autres serveurs sur Internet. Donc, il était facile de pouvoir trouver ce fichier. Il n'y avait que quelques serveurs. C'était facile à gérer. Et donc, ce fichier hôte existe toujours. C'est toujours le noyau de notre système. Le système a un peu changé, parce qu'on a maintenant des pointeurs et non pas un répertoire.

Ce fichier hôte est entretenu par le centre d'information de réseau de Stanford, et si vous aviez un ordinateur connecté à Internet, l'idée était que tout le monde envoyait ses informations au SRI, à l'Institut de recherche de Stanford, pour que tout le monde ait ces informations ajoutées à ce fichier hôte. Les mises à jour s'envoyaient par courriel et tout le monde téléchargeait… À ce jour, les nouvelles versions du fichier une fois par semaine, et tout est maintenu sur un serveur FTP.

Internet a grandi de plus en plus dès qu'on a créé ce système. On a commencé à voir qu'il y avait certains problèmes associés à ce système. Par exemple, des conflits associés aux noms. Donc si vous vouliez un registre et un nom qui n'étaient pas disponibles, on se disait : « Bon. Mon serveur s'appellera Steve, et puis c'est tout. » Mais plus on avait de serveurs, plus on voyait un concept ou un problème, ou les personnes voulaient avoir toutes le

même nom et on n'avait pas de bonnes méthodes pour empêcher qu'il y ait des doublons. Donc, on a commencé à travailler sur le concept du DNS et sur le concept d'Internet comme un système d'identificateur unique, ce qui a presque immédiatement mis fin à l'ancien modèle.

À l'époque, on avait également des problèmes de synchronisation, parce qu'il était optionnel de télécharger le fichier hôte mis à jour et c'était le gérant du réseau de chaque société qui devait accéder au site de SRI et décider de télécharger la dernière version du fichier, c'est-à-dire qu'il n'y avait pas de synchronisation au niveau du fichier. Il était probable que tout le monde ait différentes versions du fichier dans ces ordinateurs clients. Donc, on n'avait pas de synchronisation des données elles-mêmes, c'est-à-dire que certaines personnes pourraient ne pas avoir à télécharger la dernière version du fichier et qu'ils auraient des anciennes informations, que les serveurs aient changé depuis leur dernier téléchargement ou alors que leurs informations n'aient pas été ajoutées au fichier qu'il avait. Donc, ils n'avaient pas les données complètes.

Le trafic et la charge ont beaucoup changé aussi. De nos jours, on a des connexions par fibre optique, on a des larges bandes partout. Mais à l'époque, on se connectait à travers des modems de dial up et on avait entre 200 et 1200 BPS de largeur de bande.

C'était bien d'avoir une connexion Internet, mais c'était très, très long. Ça prenait du temps de télécharger ce fichier, non seulement pour l'utilisateur, mais pour le SRI, cela impliquait une grande charge aussi parce qu'il y avait beaucoup de personnes qui le téléchargeaient en même temps.

Et le fichier hôte maintenu de manière centralisée ne pouvait pas évoluer. C'était difficile de pouvoir suivre les changements partout et d'avoir toutes les données dans un même endroit. À partir de ce moment-là, au début des années 80, on a commencé à avoir des délibérations pour remplacer ce système, pour pouvoir faire évoluer le système du fichier hôte, et pour pouvoir simplifier ce routage par courriel des informations.

Le système de courriel ou de courrier électronique était de plus en plus utilisé sur Internet aussi à cette époque. Donc, on essayait de trouver d'autres mécanismes pour faire parvenir les courriels au bon endroit. Pour ce faire, il fallait avoir de nouveaux mécanismes qui changent la méthodologie de résolution. À travers ces délibérations en matière de génie Internet, on a abouti à la création du système de noms Internet. L'IETF, l'équipe de travail de génie Internet, a beaucoup travaillé. Ils ont commencé à élaborer des normes ouvertes à travers des RFC. Ce sont des documents standard qu'ils publient. On a ici quelques exigences qui ont été établies dans leur RFC. Le RFC 799 et le RFC 819 qui étaient parmi les

premières normes à définir ce qui était le système des noms de domaine.

Plutôt que d'avoir une source centrale, une source de données centrale, comme on avait dans l'ancien modèle du SRI, on a discuté de la possibilité d'avoir une base de données distribuée qui était un meilleur modèle, plus adaptée et plus gérable aussi. Parce que c'était plus facile de gérer les différentes zones pour assurer la résolution, pour avoir la résolution sous peu. Mais l'opérateur de la zone ou le propriétaire de la zone était chargé d'entretenir son fichier.

C'était plus simple. Ça prenait moins de travail et ça impliquait moins de risque, parce que les personnes pouvaient modifier leurs propres fichiers de zone sans devoir passer par une entité centrale pour pouvoir envoyer ces modifications.

Ils utilisaient le système de cache également pour améliorer la performance et ils ont également optimisé le système, parce qu'ils fournissaient des itérations ou des répétitions pour pouvoir faciliter la redondance et la distribution de charge de ce système. Donc ici, on a un aperçu du DNS.

On parle ici de différents aspects lorsqu'on parle du DNS. On commence par la droite. On a les serveurs de noms faisant autorité. Et si on les considère du point de vue du TLD, ce sont les .COM, les .NET, .UK, .DK, par exemple. Tout ça est hébergé

dans des serveurs de noms faisant autorité. Si vous avez un TLD qui est enregistré, moi, j'ai mon propre gTLD et il est hébergé dans un de ces serveurs. Il y a d'autres serveurs de noms.

Ce sont les serveurs de noms récursifs qui fonctionnent au sein des ISP, ou si vous avez un serveur, vous pourriez également avoir ce type de serveur qui fonctionne avec les proxys et qui agissent au nom des individus au moment d'aller chercher des informations liées aux noms de domaine.

D'autre part, on a des résolveurs qui sont partiels. Ça pourrait être dans votre ordinateur personnel, dans votre téléphone portable. Tout portable qui fait une requête DNS sera également un résolveur partiel qui doit comprendre comment envoyer la requête et comment traiter la réponse. Et ça a également un niveau de cache.

Les serveurs récursifs et les serveurs partiels ont tous les deux un niveau de cache qui se résout au niveau du serveur de nom faisant autorité, mais on a différents types de serveur dont nous allons discuter au moment de voir la structure du DNS.

La base de données du DNS est un arbre inversé comme on l'appelle. C'est l'espace des noms, et si on le renverse, vous allez voir que la racine ou la partie principale commence à se diviser. C'est ce qu'on fait pour résoudre chaque noyau. Chaque noyau

est spécifique, chaque nœud est spécifique, et tout est centralisé dans un seul point de distribution.

Ici dans chacun de ces nœuds, on a un LDH, un LDH qui comprend des lettres, des chiffres ou des tirets. Donc ce sont des caractères ASCII qui sont les seuls qui peuvent habiter dans l'espace du DNS. Donc, même si on a des IDN de nos jours, c'est-à-dire des noms de domaine internationalisés, au moment d'arriver au protocole du DNS lui-même, il est nécessaire de traduire cette adresse d'IDN en caractères ASCII ou LDH.

On a un exemple d'IDN d'ailleurs ici à gauche. Vous voyez xn-j6w193g. Il s'agit d'un mécanisme de traduction que les IDN et Punycode utilisent. Ici, on a une langue autre que l'ASCII qui représente un nom de domaine d'IDN. Je ne sais pas lequel, mais si vous travaillez avec les caractères d'IDN en arabe ou dans d'autres scripts non latins, ça se représente comme cela avec ces valeurs xn.

Les étiquettes comportent un maximum de 63 caractères. C'est le maximum. Du point de vue humain, c'est déjà assez, parce que le but du DNS est de pouvoir être rappelé. Si on essaie de se rappeler des noms de domaine de plus de 63 caractères, c'est plus difficile. Et il n'y a pas de différences entre majuscule et minuscule.

Au moment de voir les différentes étiquettes, vous allez voir qu'on commence à construire des noms de domaine. On construit les noms de domaine du bas vers le haut. Donc, on a ici www.exemple.com et on verra par la suite le point caché ou le nom de domaine pleinement qualifié sous peu.

Par exemple, si on prend le cas d'exemple.com, c'est ça la structure que vous allez voir. On peut voir la structure du DNS dans le système du nom de domaine, et vous voyez que c'est lié à cette structure de l'arbre des noms de domaine. Donc, vous voyez clairement quelle est la structure des noms de domaine.

Un nom de domaine complétement qualifié veut dire que ça identifie le nœud sans ambiguïté, c'est-à-dire que dans cet exemple, par exemple, on a www.exemple.com, personne ne va saisir le point à la fin du COM. Mais ici, on va écrire com., et c'est un système qui est complétement qualifié : notre nom de domaine est complétement qualifié. Je ne veux pas qu'il y ait d'incertitude par rapport à ce nom de domaine. Le nom de domaine reconnait ce point à la fin comme l'autorité du nom de domaine comme un nom de domaine complétement qualifié. Ça veut dire qu'on s'attend à accéder exactement à ce nom de domaine. La plupart des noms de domaine ne sont pas ambigus. Donc, on a tendance à ne pas ajouter le point à la fin, mais si on ne le fait pas d'habitude, c'est le système lui-même qui va ajouter ce point.

Un domaine est un nœud et tout ce qui suit au-dessous. Donc, le nœud principal d'un nom de domaine implique que tout ce qui est au-dessous du nom de domaine fait partie de ce domaine. Donc, en haut de l'arbre, ce sommet du nœud, on doit avoir des pointeurs qui vous permettent de gérer le reste de l'espace ou de vous renvoyer au reste de l'espace. Ça va vous donner des pistes de comment arriver au niveau suivant ou au point suivant. On verra ça au moment de voir les exemples de résolution.

Par exemple, si on prend ça et qu'on arrive jusqu'à la racine, la racine sera le sommet de tout le système. Si on prend le ccTLD .UK, il aura des noms de domaine sous le UK, et le UK sera le sommet de ce système de noms de domaine.

L'espace de nom est divisé en un modèle de distribution qui permet d'assigner les différentes identités qui gèrent les fichiers de zone ou le nœud. Le nœud, c'est ce qu'on appelle une zone et cette distribution de zone nous permet de réduire la gestion en un seul point, à la consolider en un seul point et à obtenir des réponses plus rapides au moment d'apporter des modifications.

Donc si vous êtes une société, par exemple, pour le cas du .COM, et si vous avez plusieurs serveurs, plutôt que d'envoyer un courriel à votre opérateur de registre. Si vous êtes l'administrateur d'exemple.com et que vous voulez apporter des modifications à votre propre zone, ça va se propager partout sur

Internet. C'est plus facile d'envoyer des changements à une seule zone. C'est plus rapide.

Les noms de domaine sont délégués par les fichiers parents. Moi, j'ai compté .NET, par exemple. Et la partie qui fait la délégation de mon nom de domaine est .NET. La relation parent-enfant implique que c'est moi qui gère mon nom de domaine conte.net. Moi, en tant qu'enfant, mais il y a toujours ce type de rapport parent-enfant jusqu'au moment où on arrive à la racine, parce que la racine est le parent et elle n'a aucun parent au-dessus.

Ici, vous voyez l'espace des noms encore, et on voit ici les limites administratives. Dans le modèle distributif, chaque partie ou chaque niveau a différents types de distribution et différentes règles pour savoir qui peut gérer ou exploiter ces noms de domaine ou de zones. Si on commence au-dessus, on a la zone racine, au sommet. IANA et PTI, vous le savez, gèrent la zone racine en collaboration avec les gérants de zones racine de VeriSign aussi. On a par la suite les TLD, les noms de domaine de premier niveau, qui peuvent comprendre les ccTLD, c'est-à-dire des extensions géographiques, des extensions génériques, des gTLD, tout autre domaine de premier niveau qui existe.

Et puis en dessous, au niveau suivant, on a exemple, foo et bar. Ce sont des noms de domaine, et chacun est exploité par un titulaire de noms de domaine. Et puis, au niveau suivant, on a

d'autres entrées : ça pourrait être un serveur, un sous-domaine, mais tout cela est géré dans la limite administrative du nom de domaine. Ici, vous voyez le rapport parent-enfant et la délégation, à chaque fois.

Les serveurs de nom maintenant sont ceux qui répondent aux requêtes. Une requête est envoyée par une application ou un autre serveur de nom ou un autre résolveur. Par exemple, on a vu qu'il y avait des résolveurs partiels, il y avait des résolveurs de cache. Et un serveur de nom peut être un résolveur et à la fois être un serveur de nom aussi. Un serveur de nom faisant autorité à ses propres informations de zone pour ce nom de domaine, c'est-à-dire que c'est la source faisant autorité pour ce nom de domaine. Ce serveur de nom faisant autorité à toutes les connaissances de la zone et c'est là que l'opérateur d'un nom de domaine va ajouter ses informations.

Un serveur de nom faisant autorité peut fournir des réponses définitives aux requêtes sur la zone. Les zones peuvent avoir différents serveurs faisant autorité qui vous donnent une redondance et qui partagent la charge de requête. Plus on a de serveurs dans une zone, plus on distribue la charge de manière topographique dans le réseau, et mieux on distribue la charge.

Dans ce système, on a d'habitude un serveur de nom qui est le maître et puis on a des serveurs esclaves ou secondaires. Le

serveur maître ou primaire est celui qui reçoit les changements aux données de zone de la part de l'opérateur du nom de domaine. Donc, si j'ajoute un nouveau nom de domaine ou que je change une adresse IP, par exemple, mes modifications seront présentées au serveur maître ou primaire.

Les serveurs secondaires vont répéter exactement ce que fait le primaire, mais ce sera fait à travers un processus de requête et réponse, c'est-à-dire que les modifications seront envoyées au serveur primaire. Une fois qu'elles seront demandées par les serveurs secondaires, ils vont recevoir les mêmes informations que j'ai envoyées au serveur maître et répéteront ces mêmes informations.

Une fois que ces données de zone sont distribuées entre les différents serveurs faisant autorité, on voit les mêmes données partout. Le résolveur n'est pas intéressé pour savoir quel est le serveur de nom primaire. Ils font tous autorité, peu importe s'ils sont primaires ou secondaires.

On a un transfert de zone qui est initié par le serveur secondaire. On autorise les modifications au niveau primaire et on autorise… On identifie les serveurs secondaires qui sont autorisés, mais qui renvoient à chaque fois au serveur primaire.

Toutes les personnes qui veulent avoir des informations vont accéder au serveur primaire pour voir s'il y a des modifications

dans le serveur primaire. S'il y a des modifications, ces informations sont envoyées pour être mises à jour dans le serveur secondaire. Autrement, on continue comme d'habitude.

Les normes du DNS spécifient le format des paquets DNS qui sont envoyés à travers le réseau. Et cette norme spécifie également que le format du fichier maître doit être basé sur le texte.

Pour la plupart des fichiers de zone, on peut observer les serveurs primaires et secondaires, et si on a la capacité de regarder les serveurs eux-mêmes, il est possible d'accéder au fichier de zone qui n'est qu'un fichier de texte tout simple. Ça pourrait comporter quelques lignes. Mon propre nom de domaine, je n'ai quelques entrées, quelques lignes. C'est facile à lire. Mais le .COM, par exemple, a des millions de saisies, parce qu'il y a beaucoup de noms de domaine qui appartiennent à ce TLD.

Ça pourrait être un fichier très simple ou très complexe, donc en matière de taille du fichier. Ça va dépendre de la complexité des services qu'ils fournissent. Mais tout dépend en fait de la taille en définitif. Puisque c'est facile à lire, il est facile à diviser aussi, et c'était ajouté dans ce format parmi les spécifications pour les standards ou les normes du DNS.

Chaque nœud a un nom de domaine qui pourrait avoir différents types de données associées. On l'appelle des registres de source ou des archives. Nous allons en voir quelques exemples dans la diapo suivante, je pense. Une zone pourrait comprendre différents types de registres de ressources. Donc, on pourrait se dire que ces courriers électroniques sont envoyés à un tel serveur et que FTP va dans un notre serveur, le www va dans un autre, DNSSEC a son propre fichier.

On a différents types de ressources ajoutées au serveur et au fichier de zone, ce qui permet que les requêtes qui arrivent soit plus spécifiques et soit répondues plus spécifiquement aussi en matière de gestion du trafic. On ne peut jamais mélanger des fichiers de zone dans différentes zones. Donc, chaque fichier parle d'une zone spécifique.

Si vous gérez différents domaines et que vous avez différents serveurs de noms pour ces différents noms, vous pouvez le faire, mais il faut que chaque domaine ait son propre fichier pour que ces données soient mieux gérées. Donc, les types d'enregistrements de ressources ont différents champs. On a d'une part le titulaire. C'est la personne qui est associée à cet enregistrement de ressource. On a le cache, qu'on verra une fois qu'on aura vu la résolution. Le temps de validité des informations va changer et chaque domaine a des informations pertinentes.

Donc, si vous faites différentes modifications sur un serveur spécifique, le niveau de cache doit être maintenu au minimum pour pouvoir envoyer ces informations plus rapidement sur Internet. Mais si vous allez faire des modifications plus compliquées, il faut que ce soit un peu plus long. Mais pas trop long, pour que les données puissent être mises en cache. Pourtant, les données doivent trouver leur échéance à un moment ou à un autre, parce que les serveurs de noms et les résolveurs vont devoir venir chercher les modifications et une fois que ce fichier aura trouvé son échéance, et c'est ce à quoi on a intérêt.

Il y a également différentes catégories ou différentes classes. On ne les utilise pas beaucoup de nos jours, mais on a différents types de classes. Sur Internet, on a un fichier qui s'appelle IN, classe Internet. Et puis, on a différents types d'enregistrement de ressources. On a des enregistrements [inaudible], par exemple. Mais ce type vous explique ce que le fichier va contenir, et la plupart des résolutions entre les noms et les numéros se fait à partir du type d'enregistrement de ressources.

Et finalement, on a le RData, c'est-à-dire des données du type spécifié qui sont contenues dans le fichier.

Ici, vous voyez la syntaxe typique. Tout ce qui apparait entre crochet peut être éliminé. Si vous allez présenter des

modifications à un fichier de zone, tout ce qui est entre crochet peut être supprimé, et ça va reprendre des valeurs par défaut au moment de compléter tout cela. Mais ici, on a la classe, le type, les données. Vous voyez le nom de titulaire, le temps de validité des informations, classe, type et données. Typiquement pour les zones racine, la classe sera In, comme je le disais tout à l'heure, et si on ne spécifie pas une durée spécifique pour le temps de validé des informations, ça héritera ces informations du fichier ou de l'enregistrement maître pour ce fichier.

Il faut toujours mettre le type et les données, parce que c'est là qu'a lieu la résolution. Je vais vous montrer un exemple par rapport à cela et vous pouvez avoir www et après, vous pourriez avoir l'adresse IP.

Quels sont les types les plus fréquents d'enregistrement de ressources ? On a les types A, qui vont résoudre le nom en une adresse IPv4. Ensuite, on a des enregistrements de type AAAA, qui vont résoudre les adresses IPv6.

Ensuite, on a des enregistrements de ressources NS, qui vont enregistrer le nom de serveur faisant autorité et dans quelle mesure ce serveur est lié à l'autorité.

Puis, on a SOA, début d'autorité. C'est le premier enregistrement de ressource que vous allez voir dans la zone et qui va nous

raconter beaucoup de choses. D'un côté, quel est le temps de validité des informations par défaut, et d'autres informations.

Nous avons un enregistrement CNAME. C'est un alias, c'est-à-dire quand on a un enregistrement A et qu'on veut avoir… arriver à une adresse IP et qu'on voit que ce soit un serveur FTP, il faut mettre donc un CNAME pour le FTP. Et ensuite, je peux dire CNAME FTP www, et puis on introduit un alias pour le serveur.

Ensuite, des ressources MX. Elles sont notamment utilisées pour des services de messagerie. Lorsqu'on veut envoyer un message de courrier électronique, on ne sait pas quel sera le service de messagerie utilisé. Donc, on va interroger les différentes couches du serveur pour voir quel sera le serveur de messagerie qui recevra notre message. Alors, à ce moment-là, on utilise cet enregistrement de ressource MX et en réponse, on obtient une liste des adresses IP et de noms de domaine qui sont associés au service de messagerie pour ce domaine.

Ensuite, nous avons un enregistrement de ressource PTR, lorsqu'on met en corrélation un nom avec un numéro. Ce type de ressource est utilisé pour la résolution inverse, c'est-à-dire quand on va résoudre un numéro en un nom. Alors, on saisit l'adresse IP et nous obtenons le nom d'hôte. C'est utiliser un

enregistrement de ressource qui est utilisé, mais pas aussi fréquemment que les autres.

Il y a actuellement 84 types d'enregistrement de ressource selon le répertoire établi l'année dernière. La plupart d'entre eux sont des cas spéciaux. Nous allons voir notamment ceux qui sont les plus fréquents, mais il y a beaucoup d'autres types et il y a beaucoup de cas spéciaux qui ne sont pas toujours utilisés. Nous essayons de voir de plus en plus avec le DNSSEC les différents types qui apparaissent. Mais pour la plupart, en général, il y a une dizaine de noms de domaine. Il y a une dizaine, pardon, d'enregistrements qui sont les plus utilisés.

L'IANA a un type spécial d'enregistrement de ressource. Cette présentation sera disponible dans le calendrier. Je vais la mettre à votre disposition, et ici, vous allez trouver de manière plus spécifique les informations que je vous communique maintenant.

Nous savons qu'IANA s'occupe entre autres de gérer les identificateurs uniques. L'IETF gère les RFC, c'est-à-dire les normes et les normes de paramètres de protocole pour le DNS. Il faut qu'il y ait un endroit où les types d'enregistrement de ressource puissent être stockés. Et l'IANA s'occupe justement de gérer ces enregistrements de ressources, ces identificateurs uniques.

Voici un exemple de la page IANA où on voit tous les types d'enregistrements de ressources. Vous voyez au tout début les différents types, NR, A… Les différents types, vous voyez. Il y a… Nous n'allons pas rentrer dans le détail, mais il y a beaucoup de types d'enregistrement de ressources.

En général, le DNS est utilisé pour mettre en correspondance un nom et une adresse IP. On fait cela en mettant en corrélation le nom, par exemple, avec une adresse IPv4 ou IPv6 et on peut avoir dans ce cas… Vous voyez, on a une requête et on peut avoir le même nom d'hôte qui va trouver une adresse IPv4 et une adresse IPv6.

Les serveurs de noms. Il faut avoir un enregistrement de serveur de nom dans chaque zone et il faut que cela se trouve dans la zone parent. On en reparlera plus tard, mais si vous vous rappelez le parent dans l'exemple de .COM. C'est .COM, c'est le parent ; et puis l'enfant, c'est exemple.com. La seule information que le parent possède par rapport au domaine .COM est aux zones ultérieures : c'est l'enregistrement du serveur de nom et l'information DNSSEC que cela peut contenir.

Donc, la requête doit aller à cette zone pour pouvoir obtenir l'information. Or, si vous n'avez pas un pointeur d'un parent vers l'enfant, la requête ne sait pas où aller pour obtenir la réponse. C'est pour cela qu'il faut avoir ces informations dans la zone

parent, mais aussi dans la zone enfant, pour savoir où sont les autres serveurs.

Vous voyez ici, ce n'est pas une adresse IP. L'enregistrement NS est un pointeur vers d'autres domaines et c'est un nom complétement qualifié.

Comme vous le voyez, le parent dans ce cas, c'est la racine qui a des pointeurs vers les enfants. Dans ce cas .COM et dans la racine, nous avons une liste d'environ 13 gtld-servers.net, c'est-à-dire qu'il y a 13 serveurs de noms associés à .COM. Et donc ce qu'on fait, c'est pointer les requêtes vers ces serveurs.

Mais ici, il y a le problème de l'œuf et de la poule, car on a la résolution de nom et on pointe vers un nom de domaine. Mais si on n'a pas fait de résolution de nom avant et qu'on pointe vers un domaine, comment arriver à cela ? Alors, il y a ce qu'on appelle les enregistrements de type GLUE. Voilà. Alors, il y a des enregistrements de type GLUE, et donc on fait l'hypothèse que vous n'êtes pas encore allés à un domaine en particulier, que vous n'avez pas obtenu la correspondance exacte.

Alors, vous devez aller vers ces serveurs de noms, mais vous ne savez pas de quel serveur de nom il s'agit. Donc, on ajoute ces enregistrements GLUE dans les parents et le GLUE va nous dire l'adresse IP de ces noms. Alors, ici, on dit exemple.com va vers

exemple.com et donc le serveur .COM ne sait pas comment arriver à exemple.com.

Dans ce cas-là, on ajoute un enregistrement GLUE qui nous dit quelle est l'adresse IP à laquelle il faut aller pour pouvoir obtenir ce nom de domaine. Et à ce moment-là, la résolution se fait sans être jamais aller auparavant à ce serveur.

Le GLUE peut être un enregistrement A ou AAAA dans l'espace IPv4 ou IPv6. Si on utilise des serveurs v6, il est important d'utiliser ce type d'enregistrement. Nous parlons du début d'autorité SOA. Il s'agit de l'information qu'on obtient et je vous montre ici exemple.com. On a donc un début d'autorité qui nous montre quel est le serveur de nom primaire.

Et ensuite, il y a l'adresse courriel, mais ce type de symbole peut vouloir dire différentes choses dans différents endroits. Ce que nous faisons maintenant, nous changeons le point, donc point… Exemple.com, c'est hostmaster.exemple.com. De cette manière, on va changer donc la gestion de zone. Ce n'est pas toujours le cas, mais on le voit souvent.

Nous avons un nom de série qui nous permet de savoir ou de comprendre quelle a été la dernière date où on a introduit les modifications dans une séquence. Donc, dans cet exemple, nous voyons qu'il y a en 2016 le mois de mai, 1er mai 2016, c'était la

dernière date à laquelle une modification a été introduite. Les différents formats de date qu'on peut introduire…

Et le 00 est un numéro de séquence. Donc si on fait d'autres modifications au lieu d'introduire quelque chose de nouveau, on fait évoluer le numéro de séquence. Et on va donc implémenter le dernier numéro de série.

Donc, quand un serveur secondaire va interroger cela, il va comparer les numéros de série, en disant : « Voilà le numéro de série. » Si c'est moins que le numéro de série primaire, si c'est plus important le numéro est plus élevé. Cela veut dire qu'il faut aller chercher une nouvelle mise à jour. Et donc, on lance une requête pour interroger les serveurs de noms.

Il y a différentes valeurs : refresh, retry, expire, minimum. Tout cela a différentes significations dans les différentes parties. Je dirais que le cache minimum, c'est plutôt refresh. Quel est le cache par défaut, le TTL ? Vous pouvez le dire. S'il y a un TTL par défaut, c'est-à-dire un temps de validité des informations par défaut, ça devrait apparaitre ici ?

EDWARD LEWIS: Oui. Bonjour. Le TTL par défaut doit être établi dans la zone, le fichier de zone.

Il y a une petite confusion par rapport à cela. Le dernier numéro ici, c'est le défaut pour des réponses négatives, c'est-à-dire si on dit non. Cela peut durer cinq minutes. Mais il y a un défaut pour des réponses positives.

STEVE CONTE: Merci beaucoup. Merci. Il y a seulement un enregistrement de ressource SOA par zone, par fichier de zone.

Maintenant, si nous voyons le type CNAME, nous parlons d'enregistrement de type A. Le CNAME crée un alias, et si on fait l'hypothèse que nous avons déjà créé un enregistrement A pour exemple.com, mais nous avons besoin pour mail.exemple.com d'un nom de domaine différent, nous ne pouvons pas créer un autre enregistrement A et c'est pour cela que nous utilisons le CNAME. Et donc, nous mettons l'enregistrement A avant. Par exemple, mail.exemple.com et ensuite, le CNAME.

Non, excusez-moi, c'est l'inverse. Le CNAME, c'est la cible. Alors, le nom mail, c'est celui qu'on met dans l'enregistrement A et la cible, c'est le nouveau nom. Cela crée un alias et on ne met pas des alias après des alias, après des alias et des alias. On introduit des alias. On pointe toujours sur l'enregistrement A.

Quand nous envoyons un courriel, les serveurs de messagerie doivent savoir… On doit savoir comment arriver aux serveurs de

messagerie qui servent ce nom de domaine et nous savons cela. Avant, on faisait des recherches au niveau des adresses, mais il n'y avait pas de flexibilité. C'est pour cela qu'il a fallu changer les serveurs de messagerie. Autrement, cela devenait très difficile.

Le DNS offre une plus grande flexibilité en ajoutant le type d'enregistrement host. [Inaudible] Et donc, on peut spécifier le nom du serveur dans ce domaine et une préférence. Dans cette diapo, nous pouvons voir qu'il y a deux serveurs de messagerie associés à .COM. On a un enregistrement MX avec une valeur de 10 qui va à mail.exemple.com, et puis un enregistrement MX avec une valeur de 20 qui va à mailbackup.exemple.com.

Cela nous permet d'avoir plusieurs serveurs de messagerie, sauf qu'il y a une valeur qui est moins importante que l'autre. Donc moins cette valeur est importante, moins ce serveur sera le serveur préféré. Voilà ce que vous voyez sur l'écran. Mais si ce serveur de messagerie n'est pas disponible pour quelle que raison que ce soit, nous voulons toujours tenir notre serveur de messagerie. Et c'est pour ça qu'on a cette autre valeur 20. Donc, si on ne peut pas atteindre le serveur de messagerie qui a la valeur 10, on peut accéder au serveur de messagerie qui a la valeur 20.

Tout ce qui est à gauche correspond à l'utilisateur et tout ce qui est à droite correspond au domaine. En ce qui concerne la

résolution inverse, cela est utilisé par certains administrateurs pour différents services.

Ce qu'on fait ici, c'est qu'il y a un fichier de zone, qui s'appelle in-adder.arpa, et donc on établit cette zone comme une zone d'administration. Par exemple, si on a cette entrée ici 7.2.0.192.in-adder.arpa. Ce que cela fait, c'est que ça met en correspondance avec une adresse IP, mais de manière inverse.

C'est-à-dire qu'on a l'adresse IP associée à exemple.com, et si on fait cela à l'inverse, vous voyez qu'on passe donc de l'adresse… du nom à l'adresse IP. Donc, comme dans tout autre domaine, in-adder.arpa nous permet de faire la résolution inverse.

Cet in-adder.arpa est géré par IANA. Il gère cela… des parties de certaines zones de cela pour les RIR. Les RIR délèguent une partie de cet espace à leurs consommateurs. Puis, les consommateurs gèrent ces parties qui leur ont été déléguées. Donc, ce type de ressource que je vous montre est utilisé pour faire la résolution inverse. Il y en a un pour IPv6 et un pour IPv4. C'est IP6.arpa pour IPv6…

Très bien. Maintenant, les extensions de sécurité du DNS. Il y a une séance plus approfondie par rapport au DNSSEC pendant cette réunion. Si vous êtes intéressés par cette partie donc du DNS, je vous conseille de participer à cette séance. Je vais vous donner un bref aperçu de cela. Quand on pense à la sécurité du

DNS, on pense au chiffrement de données ou à des choses comme ça. Le DNS ne chiffre pas les données, et donc le…

Il s'agit plutôt d'une authentification des données, authentification de la source et de la cible ou de la destinée des données. Le chiffrement est destiné à savoir que… L'authentification, pardon, est destinée à savoir que la source et la cible sont bien la source et la cible auxquelles nous nous attendions.

Dans le DNSSEC, il y a plusieurs types de ressources. On a DNS key, c'est la clé publique de zone, et donc dans la gestion de clé, on a une clé publique et une clé privée. Le gestionnaire de [inaudible] va avoir la clé privée. Et ensuite, on publie la clé privée et à travers certains algorithmes et méthodes, nous faisons des résolutions. Nous comparons la clé publique et la clé privée pour être sûr que ce qui se passe est vraiment ce qui doit se passer. Par exemple, …

Bon. Comme je vous l'ai dit, il y a une séance plus approfondie par rapport au DNSSEC, je crois que c'est mardi, où ils vont rentrer plus en détail par rapport à comment cela fonctionne, comment fonctionne les algorithmes, quelles sont les comparaisons entre les clés publique et privée. Nous avons aussi NSEC et NSEC3 : c'est le pointeur qui va vers le prochain nom dans la zone, ce qui nous permet d'authentifier un dénie

d'existence. Ce qu'on fait, c'est de voir s'il n'y a aucun domaine qui n'existe. Par exemple, icann58.com, on veut savoir… On veut avoir une réponse autoritaire pour savoir si ce domaine existe ou non. Si on n'obtient pas de réponse, il n'y a pas de certitude que le domaine n'a pas existé.

Donc, cela nous donne une réponse qui fait autorité et qui nous dit : « Cela n'existe pas. » Et donc, on sait qu'on ne doit pas chercher ce type de domaine.

Ensuite, la signature de délégation. Cela fait partie de la chaîne de confiance. On a un enregistrement DS dans la zone au-dessus de vous. Par exemple, si on a .COM, il faudra insérer un enregistrement DS à .COM pour pouvoir construire cette chaîne de confiance. Donc quand on va passer au modèle d'authentification, cela va jouer un rôle important.

Ensuite, on a TXT, on a URL. Ce sont d'autres types d'enregistrement de ressource. Il y a d'autres types spéciaux qu'on utilise rarement ou qu'on utilise à des fins spécifiques. Par exemple, TLSA qui est utilisé par DANE. Ce sont des entités… C'est l'authentification identifiée nommée et qui est associée au certificat X509.

Ici, vous voyez un fichier de zone exemple. C'est un exemple de ce que… d'une zone… d'un fichier de zone en format texte. Nous voyons donc qu'il y a l'enregistrement SOA, des

enregistrements NS. Nous avons un enregistrement NS qui est lié à un nom de domaine complétement qualifié.

Ensuite, nous avons un enregistrement A avec une adresse IP après. Ensuite, on a un enregistrement AAAA. Ensuite, on a un MX. On a CNAME avec un nom canonique et puis, on a encore l'enregistrement NS lié à une adresse IP.

Vous voyez donc un exemple typique de ce que les utilisateurs de noms de domaine vont utiliser pour associer des noms d'hôtes à des noms de domaine. On a www, on a FTP, différents noms d'hôtes qu'on peut ajouter. Et puis, plus on rentre dans… plus on rentre au cœur de l'arborescence, plus le nom sera long.

Maintenant, nous allons passer au processus de résolution. Avant de faire cela, je vais faire une petite pause pour voir si vous avez des questions par rapport à ce que je viens de présenter, avant de passer à la résolution. Est-ce qu'il y a des questions ? Je pense que tout le monde s'endort. D'accord. Je vois une main levée. Je vais vous donner le micro. Voilà.

PERSONNE NON IDENTIFIÉE: Je pense que vous en parlerez, mais la zone… les fichiers de zone, donc… Et il y a ce type de fichier de zone dans chaque zone ?

STEVE CONTE: Dans le fichier de zone… Est-ce qu'on peut passer à la diapo précédente ?

PERSONNE NON IDENTIFIÉE: Alors donc, est-ce que ce type de fichier de zone, on le retrouve au niveau de la racine, au niveau de .COM ?

STEVE CONTE: Ce fichier de zone correspond à exemple.com, mais il y aura un fichier de zone pour .COM et il y aura un fichier de zone pour la zone racine. Et s'il y avait des sous-domaines dans exemple.com, il y aurait un fichier de zone. Par exemple, gouv.exemple.com. Si on veut que cette délégation soit gérée par une autre entité, on peut déléguer cela à ce niveau et pointer un enregistrement NS vers un sous-domaine. Est-ce qu'il y a des questions en ligne ?

[CATHY]: Nous avons une question en ligne de Jared. Il veut savoir si nous allons nous pencher sur les questions liées à la propriété intellectuelle.

STEVE CONTE: Non. Nous, on parle ici des processus de résolution du DNS.

Très bien. Nous allons parler de la résolution maintenant. Si on reprend le modèle qu'on a vu tout à l'heure, on avait vu qu'il y avait des résolveurs partiels, des résolveurs récursifs et des serveurs de noms faisant autorité. Avant de passer à la résolution, vous aurez vu qu'il y a deux types de requête : les requêtes récursives et les requêtes aux résolveurs partiels, c'est-à-dire que les requêtes récursives demandent des réponses complètes ou des erreurs.

Les requêtes récursives envoient des requêtes répétées aux serveurs non récursifs, aux serveurs de noms récursifs. Donc, la réponse sera : « Je peux chercher une partie et j'accepterais des recommandations d'où aller chercher. » On en reparlera.

On a des algorithmes de haut niveau, etc. Il faut voir s'il y a des coïncidences avec les données locales. Si possible, on va donner ça comme réponse. S'il n'y a pas de réponse exacte possible, on va vous envoyer chercher des données locales.

Et si c'est une requête récursive, on vous renvoie à la zone la plus proche. Vous vous rappelez qu'on avait des points autour de chaque zone, et on vous indiquera de suivre les recommandations ou les renvois partout dans l'arbre pour arriver jusqu'au bout.

Comment peut-on donc recommencer la résolution sans avoir de données locales ? Si c'est la première fois que vous accédez

ou que vous venez de créer votre fichier et que le cache est vide, et que vous ne savez pas comment faire, chaque serveur a un fichier de piste qui contient des pointeurs vers les zones racine, c'est-à-dire qu'il y a des entrées, des saisies pour les serveurs racine dans ce fichier. Donc, un nouveau serveur de nom ou un serveur de nom sans cache pourra arriver jusqu'au serveur racine, au moins à travers des raccords, à travers des enregistrements de GLUE qui vous enverront au serveur racine suivi par le serveur de nom.

Donc, on aura un registre A dans chaque zone racine et puis une adresse IP dans les serveurs racine, et cela s'applique aux 13 instances du serveur racine.

Ici, on a un échantillon d'un fichier de piste qu'on peut obtenir sur InterNIC.net, mais ce fichier est disponible dans tous les logiciels que je connais. Chaque logiciel de navigation contient un fichier de ce type et le fichier de piste ne change pas très souvent.

On a l'administration de la zone racine. C'est la personne qui exploite la zone racine. Vous allez écouter parler du RZM. C'est le mainteneur de la zone racine, la personne qui fait l'entretient, c'est-à-dire que c'est une fonction qui est divisée entre la PTI qui exploite les fonctions IANA et VeriSign qui est le mainteneur de la zone racine. Les deux travaillent ensemble pour créer et

maintenir la zone racine, et puis, on avait 12 organisations qui exploitent les fichiers de zone racine. On les a toujours.

Les opérateurs de zone racine se réunissent au sein du RSSAC. Ils seront là cette après-midi, ici. Ils auront leur présentation dans la salle C2. Ils vont expliquer ce qu'est la zone racine et comment elle interagit avec… Et, comment le RSSAC interagit avec les mainteneurs de la zone racine.

Même en sachant qu'il y a 13 instances du serveur racine, il y a moins d'opérateurs. Il y a 12 opérateurs, parce que VeriSign exploite les A et J. L'un des mythes était que la zone A faisait plus autorité que les autres au serveur racine, mais ce n'est pas vrai. Tous partagent les mêmes informations exactement.

Aucun de ces serveurs n'est un serveur primaire. Ils sont tous considérés comme des serveurs secondaires ou esclaves. Ils utilisent une méthodologie qui s'appelle le maître caché, c'est-à-dire qu'il y a un serveur qui ne communique qu'avec les adresses IP ou les instances de ces serveurs racine. Ils agissent tous comme des serveurs secondaires et ils cherchent des données dans le serveur primaire qui est caché. Le serveur maître caché ne communique qu'avec ces serveurs racine. C'est fait de cette manière, pour que s'il y avait un manquement à la sécurité de ce serveur primaire, les données de zone, de fichier de zone faisant

autorité, ne seront pas publiées sur Internet, mais il sera toujours protégé.

Si jamais il y avait une de ces machines qui serait compromise, ce serait seulement la machine ou le serveur qui aurait été compromis et non pas tout le système de serveur racine. Si la racine, elle, par exemple, avait des problèmes de sécurité et que le conte.net – je ne sais pas qui voudrait le kidnapper, mais peut-être qu'on voudrait me kidnapper mon nom de domaine, si quelqu'un voudrait trouver les données pour le conte.net, seulement le serveur L publiera ces données, parce que ce n'est pas un serveur faisant autorité, c'est-à-dire ce n'est pas des données de zone primaire. Et mes données sont publiques jusqu'à ce qu'on me change mon numéro de série ou jusqu'à ce que quelqu'un se rende compte que conte ne devrait pas être inclus dans ce fichier. Mais au moment de changer les numéros de série, on se rendra compte que les numéros de série n'ont pas été mis à jour et on demandera au fichier maître d'envoyer les nouvelles données pour que tout fonctionne correctement.

Les serveurs racine ont chacun leur propre site. Le site principal pour tous les serveurs est root-servers.org. On les appelle instances. Chaque serveur est une instance de la racine. On verra davantage de détails là-dessus. Mais il y a plus de 13 serveurs racine aujourd'hui à travers une technologie qui s'appelle Anycast, dont on ne discutera pas aujourd'hui. On a la

possibilité de répéter une instance du serveur racine avec la même adresse IP pour envoyer des annonces, par exemple.

Donc, ces 13 serveurs racine sont devenus des centaines d'instances du serveur racine qui partagent ces informations autour du monde. C'est vraiment fantastique, parce que ça nous permet d'assurer différentes fonctions. D'une part, ça partage la charge et plutôt que d'avoir 13 serveurs qui acceptent la charge de tout le système Internet au niveau de la racine, on a maintenant des centaines d'ordinateurs qui le font. Ça équilibre cette charge, parce que les instances de la racine sont diverses au niveau mondial. Elles sont partagées partout dans le monde. Elles se partagent la charge d'Internet dans les différentes régions du monde et c'est un système très, très robuste qui nous protège également des attaques DDoS. Grâce à la technologie Anycast et à la manière dont elle fonctionne, le système est plus résilient, plus élastique, et il peut faire face de manière plus souple aux différentes attaques DDoS, des attaques de reflux de service.

Si vous accédez à root-servers.org, vous allez voir la carte avec les différents points verts et jaunes qui vous indiquent combien d'instances sont présentes dans chaque emplacement. Ici, on voit le processus de changement à la zone racine. Dans ce cas-là, le gérant ou l'opérateur du TLD peut être n'importe qui : le .COM,

le .UK. Il s'agit d'un gérant, d'un opérateur de nom de domaine de premier niveau.

Qui peut présenter des changements aux données de son nom de domaine au niveau de la racine ? Ce que fera cet opérateur, ce sera d'envoyer des informations liées aux changements à l'opérateur des fonctions IANA qui traitera ces modifications pour s'assurer que la demande est présentée par une source faisant autorité. Dans le cas spécifique de la zone d'où la demande vient, donc on ne peut pas apporter une modification au .DK en tant que .UK, par exemple. Donc, l'opérateur des fonctions IANA suit des étapes pour vérifier que la personne qui demande le changement ait l'autorité pour le faire, qu'elle a la compétence pour le faire. Donc, il y a des méthodes externes et internes pour s'assurer que c'est bien la personne qui peut le demander.

Une fois que cela a été fait… une demande de mise en œuvre et envoie cette demande de modification des informations au mainteneur de la zone racine, le mainteneur de la zone racine saisira ces modifications dans une base de données qui génère par la suite un fichier de zone racine, qui est ensuite intégré au maître caché, au fichier maître caché. On pourrait l'appeler le serveur de distribution de zone, par exemple et pas le maître caché. Mais comme je l'ai dit, donc ces informations sont

ajoutées là et ce fichier est ajouté aux requêtes qui doivent être envoyées avec les modifications.

Donc une fois par jour les serveurs racine demandent au serveur principal ou primaire s'il y a des modifications et ils téléchargent les nouveaux fichiers avec les informations modifiées. Lorsque la base de données génère donc ce fichier de zone avec les modifications, il va s'assurer qu'il y a un numéro plus neuf que ce qui est présent sur Internet. Donc, ils font augmenter les chiffres pour le numéro de version de série des fichiers.

Ici, on voit un téléphone qui a été configuré pour envoyer des requêtes à un serveur de nom récursif dont l'adresse est 4.2.2.2. On dirait que le téléphone est connecté à travers Verizon et qu'il peut envoyer une requête. Les services de Verizon utilisent un serveur de nom récursif dans leur propre réseau, c'est-à-dire que Verizon représenterait ici le FSI pour le téléphone.

Dans ce cas-là, le téléphone est le résolveur partiel ; c'est celui qui fait la requête et qui l'envoie au www.exemple.com. Il veut accéder à ce site web, mais il n'a jamais accédé à ce site web. Il n'a pas de cache et donc il ne sait pas comment y accéder. Il lui faut poser cette question.

Le téléphone a sa configuration IP, qui a des pointeurs au serveur DNS.

Si vous regardez vos ordinateurs, par exemple, et que vous accédez à la configuration de réseau, vous allez voir : adresse IP, masque de sous-réseau. Puis, vous allez voir les adresses IP des serveurs de noms. C'est comme ça que votre ordinateur sait comment accéder aux serveurs de noms récursifs pour aller trouver des réponses.

Donc, le téléphone portable accède au serveur de nom récursif et lui demande quelle est l'adresse IP du www.exemple.com. Dans cet exemple, on suppose que le serveur de nom est tout neuf et qu'il n'a jamais accédé à Internet non plus. Donc, ce serveur de nom récursif n'a pas de cache et pas de données, outre sa piste, son fichier de piste. Donc, le serveur de nom récursif va dire : « Non, je ne sais pas comment y accéder, mais je sais arriver à la zone racine, au serveur racine. » Donc, on va demander au serveur racine comment pouvoir accéder à exemple.com, et puisque je suis un serveur de nom récursif, je ferais cette demande en votre nom.

Donc, le serveur de nom récursif accède au serveur récursif et va demander quelle est l'adresse IP pour exemple.com, et le serveur racine va dire : « Je ne sais pas, mais je sais comment arriver au serveur .COM. » J'ai le registre NS pour les serveurs .COM, donc demandez-leur comment faire. Donc, le serveur racine va donner au serveur de nom récursif l'adresse des serveurs du .COM.

Le serveur de nom récursif va dire : « Très bien, donc on va aller demander aux serveurs .COM. » Il va s'adresser aux serveurs de .COM et il va leur demander quelle est l'adresse IP d'exemple.com. Et le serveur TLD va dire : « Je ne sais pas, mais je sais comment arriver à exemple.com et ces serveurs de noms, donc demandez-leur. Voici l'adresse IP des serveurs de noms d'exemple.com qui seront sans doute comment arriver à www.exemple.com. » Le serveur de nom récursif dira : « Très bien. Donc à partir de cet exemple, je vais demander aux serveurs d'exemple.com comment trouver www.exemple.com. » Donc, il va envoyer cette requête.

Et le serveur de nom exemple.com va dire : « Ah moi, je suis la source faisant autorité pour ces informations. Je vais vous donner l'adresse IP du site www.exemple.com. » Donc « voici l'adresse IP ou toutes les adresses IP associées www.exemple.com ». Je parle ici de toutes les adresses IP, parce qu'il pourrait y avoir également des adresses AAAA.

Toutes ces informations sont envoyées aux serveurs de noms récursifs qui vont dire : « Super, merci. Je vais remettre cette information à la personne qui m'a envoyé la requête. » Donc on envoie au résolveur minimum les informations pour arriver à www.exemple.com. Ça envoie ces informations à l'application, ici c'était Safari, et à travers cette adresse IP, ça permettra au portable de communiquer directement avec le serveur.

Ici, on a parcouru différentes étapes, au moins cinq étapes, cinq serveurs différents pour aller trouver la réponse. C'est assez long, mais sur Internet, ça pourrait prendre 200 millisecondes. Ça ne prend pas très longtemps de pouvoir compléter le processus, mais ce qui se passe, c'est qu'à travers les caches on peut retenir certaines des réponses clés qu'on a obtenues dans le passé.

Parce qu'on pourrait non seulement vouloir accéder à exemple.com, on pourrait vouloir accéder à nike.com. Mais ce n'est plus la peine d'aller demander au serveur racine quelle est l'adresse du .COM, parce que votre application, votre résolveur, aura déjà ces informations dans le cache. Donc, les caches vous permettent d'accélérer le processus. Ou, pourquoi pas, dans cet exemple, vous voyez un utilisateur de téléphone portable qui était connecté à travers le Verizon, qui avait les services d'Internet de Verizon. Il pourrait y avoir d'autres utilisateurs de Verizon qui veulent accéder au même site web, mais le serveur de nom récursif retient ces informations pendant une période de temps. Donc, s'il y a un autre utilisateur de Verizon qui veut accéder au même site web dans ce TTL pendant donc ce temps de validité des informations, il n'y aura plus d'autres requêtes. Parce que le serveur de nom récursif se dira : « Bien, j'ai ces données qui faisait autorité et que j'ai retenu jusqu'au moment

où ces données auront expiré. Donc puisque je l'ai, elles sont toujours valides et je les envois directement. »

On a déjà vu l'exemple de comment accéder à www.exemple.com, mais ici on veut accéder à ftp.exemple.com. On saisit cela dans le navigateur. Le navigateur va accéder au résolveur minimum et il enverra cette requête au serveur de nom récursif. Il demande comment accéder à ftp.exemple.com. Comme on l'a déjà dit, le serveur de nom récursif sait déjà comment arriver à la racine, comment arriver au .COM et comment arriver à exemple.com. Donc, ce serveur a déjà enregistré toutes ces réponses.

Donc maintenant, ce n'est plus la peine d'aller à exemple.com et de demander comment arriver à ftp.exemple.com, et exemple.com va rendre ces informations, lui répondre avec ces informations au serveur de nom récursif qui enverra ces informations au serveur minimum, qui les enverra à l'application, c'est-à-dire Safari qui pourra par la suite utiliser cette information pour communiquer directement avec le serveur. On n'a pas été forcé de passer par tous les autres serveurs.

On n'a pas communiqué avec le serveur racine ni avec le serveur TLD, ce qui représente moins de temps pour l'utilisateur, moins de largeur de bande pour l'opérateur racine ou pour l'opérateur

de TLD. On ne dirait pas que c'est beaucoup pour un opérateur lorsqu'il s'agit de bits, mais puisque sur Internet, les requêtes ne sont pas tout simplement pour les personnes et qu'il y a des machines qui envoient des millions de requêtes par seconde au niveau de racine.

On a des centaines de millions, même des milliards de requêtes par seconde. Donc, plus on peut éviter d'accéder au serveur racine pour envoyer des requêtes, mieux cela est pour le serveur racine et pour les utilisateurs aussi. Y a-t-il des questions liées à la résolution ?

PERSONNE NON IDENTIFIÉE: Bonjour. Donc si j'achète un nouveau nom de domaine, par exemple myname.com, est-ce que ce nom de domaine www.myname.com sera ajouté dans le serveur de racine indiquant qu'il faudrait que les personnes accèdent à la compagnie qui m'a vendu mon nom de domaine pour accéder à mon site web ?

STEVE CONTE: Cette question est intéressante. Donc la question est : si vous achetez un nouveau nom de domaine, comment ou alors dans ce cas-là, qu'est-ce que fait le registre ? Très bien. C'est un bon exemple d'où on est.

Vous, vous achetez un nom de domaine. Vous devenez donc le titulaire du nom de domaine. Le bureau d'enregistrement est la société qui vous a vendu ce nom de domaine. Ça pourrait être GoDaddy ou quelqu'un d'autre, pourquoi pas, oui. L'opérateur de registre sera le TLD, le domaine de premier niveau dans ce cas-là. Donc, si vous avez myname.com, l'opérateur de registre serait .COM, c'est-à-dire VeriSign.

Pour qu'Internet puisse voir vos données de zone, il faudra que vous ayez un serveur de nom de domaine faisant autorité, qui héberge votre nom de domaine myname.com.

En outre, il faudra que vous créiez des pointeurs au sein du .COM qui envoient, renvoient à myname.com. En général, cela se fait à travers le bureau d'enregistrement. Lorsque le bureau d'enregistrement vous cède le nom de domaine, on vous donne la possibilité de gérer votre nom de domaine. Puis, par la suite, les serveurs de nom que vous avez créés seront communiqués avec une adresse IP au sein du bureau d'enregistrement qui créera les enregistrements en votre nom. Or, c'est à vous de décider comment ça va se passer. Vous pouvez créer vos propres serveurs de noms de domaine faisant autorité si vous êtes une société, par exemple, et les gérer vous-même. Ou alors, si vous voulez avoir recours aux services de quelqu'un d'autre, il y a différentes sociétés qui fournissent ce type de service

d'hébergement, mais aussi de fourniture de service DNS, par exemple.

Donc, vous pouvez accéder à une combinaison de services, acheter le serveur de nom d'un revendeur, l'adresse de courriel d'un autre et avoir différents fournisseurs pour les différents services. Mais tant qu'il y a un serveur de nom qui renvoie les utilisateurs à mynama.com, tout le reste peut être intégré à la même zone racine et au même fichier de zone. C'est bon. Une autre question ?

PERSONNE NON IDENTIFIÉE: Merci. Sur les 12 organisations qui gèrent les différents services du serveur racine, quelle est celle qui a le plus de requêtes ou celle qui reçoit le plus de requêtes et qui a le plus de serveurs distribués ?

STEVE CONTE: Donc, qui reçoit le plus de trafic et quelle était l'autre question ? Et quelle est celle qui a le plus d'instances ? D'accord. Celle qui a le plus de trafic, c'est une bonne question. Je ne sais pas. On pourrait peut-être demander quelle est celle qui a le plus de trafic par région. Parce que certaines des questions sont stratégiquement plus représentatives dans des régions qui sont moins desservies, par exemple.

Il y a certaines années : l'Afrique, l'Amérique latine, l'Asie ou des régions… des sous-régions en Asie avaient des liens satellites pour accéder aux plus grands réseaux qui existaient et accéder au serveur racine. Cela était cher, parce que ça représentait un trafic à travers des satellites et ça prenait du temps aussi. Parce que topographiquement et géographiquement, ça impliquait d'accéder à différentes régions. À travers les instances et l'utilisation d'Anycast, ces serveurs racine ont pu être rapprochés des utilisateurs finaux pour ne pas avoir ces délais longs pour trouver des réponses.

Les points d'échange Internet ont été stratégiquement distribués dans les différents pays et ces points d'échange pourraient servir exclusivement une région spécifique, c'est-à-dire qu'ils pourraient recevoir différentes interrogations exclusivement de cette région, par rapport à la question de qui reçoit le plus d'interrogations. Je pense que ce n'est pas une question juste, parce que c'est censé être distributif et dynamique Donc, s'il y a des serveurs qui ne fonctionnent pas ou que le routage fonctionne mieux dans un sens que dans un autre, un jour, les quantités d'interrogations vont varier en fonction de cela.

Or, qui a le plus d'instances ? Il faudrait que vous demandiez cela au RSSAC cette après-midi, mais je pense que la racine, elle, est celle qui a le plus d'instances en ce moment. Mais je n'en

mettrais pas la main au feu. Mais il y a une quantité de serveurs racine qui fonctionnent et qui travaillent beaucoup pour essayer de faire augmenter leur quantité d'instance. Ce n'est pas une course. Ce n'est pas une compétition. Ça ne veut pas dire que les serveurs vont être plus robustes s'il y a plus d'instance, mais c'est juste qu'il faut essayer de trouver des ressources disponibles pour pouvoir distribuer ces zones dans d'autres régions. Mais toutes les données sont les mêmes, que ce soit pour la zone racine L ou K, ou A. Toutes les racines ont les mêmes informations.

Y a-t-il d'autres questions concernant la résolution ? Non. Cathy, est-ce qu'on a des questions en ligne ? Bien. Comme je l'ai dit, on a donc différents niveaux d'interactions humaines et les organismes qui existent dans le système des noms de domaine. Donc, on a différentes parties prenantes. On a un titulaire de nom de domaine ; ça pourrait être ma mère ou une personne individuelle. Le titulaire de nom de domaine est la personne qui va enregistrer son nom de domaine à travers un revendeur ou à travers un bureau d'enregistrement directement avec qui le revendeur a aussi un accord.

Une fois que le nom de domaine est enregistré à travers le bureau d'enregistrement, il aura un dialogue pour envoyer au serveur de nom des informations au fichier de zone à travers le bureau d'enregistrement et à travers l'opérateur de registre.

L'opérateur de registre est le nom de domaine de premier niveau dans ce cas-là.

Si on regarde les opérateurs de registre, on voit qu'ils desservent, ils assurent différentes fonctions. D'une part, et ce qui est le plus important, les opérateurs de registre gèrent les données opérationnelles pour cet espace de nom pour ce domaine de premier niveau. Ils ont les données pour tous les autres noms de domaines enfants en-dessous.

Donc pour domaine.com, conte.net, exemple.dk, tout cela sera exploité par l'opérateur de registre. Mais outre la gestion des serveurs de noms faisant autorité pour l'espace, ils vont également fournir d'autres services. Ils ont des informations liées au WHOIS qui pourraient être hébergées au niveau de l'opérateur de registre ou au niveau du bureau d'enregistrement en fonction de ce qui a été accordé au moment d'enregistrer un nom de domaine.

Il y a également une interface d'utilisateur sur Internet pour ces informations. Il y a également des interfaces RDAP. Le RDAP est une autre méthodologie pour trouver des informations. Et puis, donc, tout ce que je viens d'expliquer serait public, en dialogue avec le nom de domaine récursif, le serveur de nom récursif. Et puis, il y a également des aspects privés avec des fichiers de zone pour les TLD. Il y aura également des données de clients

qui seront appliquées. Ça pourrait ne pas être la même base de données. Il pourrait y avoir d'autres bases de données impliquées ici.

Il pourrait y avoir des API ou des rapports, des raccords entre les bureaux d'enregistrement, et des contrats entre le bureau d'enregistrement et l'opérateur de registre. En général, on utilise ici des protocoles EPP pour pouvoir transporter les données entre bureau d'enregistrement et opérateur de registre.

Et puis, il y a également une relation unidirectionnelle avec le titulaire de nom de domaine. Donc, si mon Internet ne me déconnecte jamais, je ne contacterais pas le .NET. Mais si ce n'est pas de ma faute et que mon nom de domaine est hors service parce qu'il y a un problème, c'est très souvent au niveau du titulaire de nom de domaine qu'il y a des problèmes de configuration.

Mais si ce n'est pas le cas, il faudrait qu'on communique avec le bureau d'enregistrement. En tant que consommateur, c'est le bureau d'enregistrement qui est mon point de contact avec l'espace de nom. On communiquera avec eux pour essayer de résoudre ce problème au niveau du nom de domaine. Je n'arrive pas à penser à des situations individuelles ou spécifiques dans laquelle moi, en tant que titulaire de nom de domaine, je

contacterais l'opérateur de registre. Attendez, il y a quelqu'un ici qui a un exemple.

PERSONNE NON IDENTIFIÉE: Pardon. Je suis un peu enrhumé. Moi, je viens du .DK et on a un opérateur de registre unique. Donc si le titulaire de nom de domaine a un problème, il nous contactera directement et non pas le bureau d'enregistrement.

STEVE CONTE: Est-ce que vous agissez en tant que bureau d'enregistrement aussi pour vos services ou vous avez des bureaux d'enregistrement qui vendent des noms dans l'espace .DK ?

PERSONNE NON IDENTIFIÉE: On a des bureaux d'enregistrement qui vendent des noms de domaine. On ne peut pas enregistrer un nom de domaine directement avec nous, mais nous fournissons des services de noms de DNS. Donc, c'est nous qui faisons le support technique des noms de domaine. Donc, c'est à vous de nous contacter si vous avez des problèmes.

STEVE CONTE: Très bien. Donc voilà un exemple. Merci, c'est intéressant. Est-ce que vous recevez beaucoup de titulaires de nom de domaine avec ce type de problèmes ?

Et quelle est la taille de votre opérateur de registre au total ?

PERSONNE NON IDENTIFIÉE: Un million trois cent mille.

STEVE CONTE: Donc, c'est assez petit comme interaction avec les titulaires de nom de domaine lorsqu'on le compare avec la quantité de noms de domaine totale. Merci. Est-ce qu'il y a d'autres questions ?

Nous devons finir à 12 h 30. Très bien. On vient de finir. Très bien. Est-ce que quelqu'un a des questions ? Nous n'avons pas beaucoup de temps, mais nous pouvons consacrer quelques minutes à des questions ou à des commentaires.

Notre prochaine séance ne sera pas dans cette salle, mais dans la salle C 1.3. À 13 h 45, on aura une séance sur le réseautage sur Internet.

PERSONNE NON IDENTIFIÉE: Je regarde… Je viens d'Afghanistan et il n'y avait pas de serveur racine dans mon pays, mais dans des pays voisins. Ma question est la suivante : comment nous créons des serveurs racine ? Comment est-on qualifié pour créer des serveurs racine ? J'ai vu que certains pays ont plusieurs serveurs racine. Comment est-ce que cela fonctionne ?

STEVE CONTE: Il faut penser que la topologie des réseaux ne correspond pas toujours à la géographie réelle du monde. Parce qu'il n'y a pas de serveur racine dans votre pays, cela ne veut pas dire qu'il n'y a pas un autre serveur racine qui est topographiquement proche de votre pays. Les réseaux ne connaissent pas de limites géographiques.

Cependant, pour répondre à votre question directement, si vous pensez que votre pays peut être mieux servi par un réseau qui soit installé dans votre pays et par un point d'échange de réseau dans votre pays, vous pouvez contacter le RSSAC. Vous pouvez participer à la séance du RSSAC de ce soir. Vous pouvez les contacter directement et leur manifester votre inquiétude. Pourquoi vous pensez qu'il y aurait le besoin d'avoir un réseau ou un serveur racine dans votre pays, ou dans la topologie ? Ou, par exemple, vous pensez qu'il faudrait qu'il y ait un IXP dans votre région. Et à ce moment-là, ils vont pouvoir vous informer si cela est faisable ou non, si cela est utile ou non.

Parce que si vous avez un serveur racine à côté de chez vous, cela n'a pas de sens d'en installer un autre. Cela fonctionne plutôt par région où il y a différentes instances des serveurs racine. Si vous pouvez justifier le bienfondé d'une telle demande, à ce moment-là, on pourra vous donner les différents

modèles et à ce moment-là, vous aurez davantage d'informations. Y a-t-il d'autres questions ?

Très bien. Bon. Je vous remercie d'être ici présents aujourd'hui. Les diapos seront publiées à la fin de notre séance et notre prochaine séance concernera le réseautage sur Internet avec Jeff Houston de l'APNIC. Nous allons donc être dans la salle C 1.2. Voilà. C 1.2 à 13 h 45. Après cela… c'est assez tard. C'est bien ça ? C'est à 13 h 45.

Oui. À 13 h 45, nous avons cette séance consacrée au réseautage sur Internet. Ensuite, à 15 h 00, nous avons une séance avec John Crain pour comprendre les abus du DNS – John Crain, c'est le spécialiste de la sécurité, la stabilité et la résilience de l'ICANN – et à 17 h 00, nous avons la séance du RSSAC consacrée aux serveurs racine.

Voilà, merci à tous et bon appétit. [Applaudissements].

[FIN DE LA TRANSCRIPTION]