Groupe de travail Réseau
Request for Comments : 2308
RFC mises à jour : 1034, 1035
Catégorie : En cours de normalisation

M. Andrews, CSIRO
mars 1998

Traduction Claude Brière de L'Isle

Mise en antémémoire négative des interrogations du DNS (DNS NCACHE)

Statut du présent mémoire
Le présent document spécifie un protocole de l'Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et suggestions pour son amélioration. Prière de se référer à l'édition en cours des "Normes officielles des protocoles de l'Internet" (STD 1) pour connaître l'état de la normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Résumé
La [RFC1034] décrivait la façon de mettre en antémémoire les réponses négatives. Elle portait cependant une faute fondamentale en ce qu'elle ne permettait pas qu'un serveur de noms ressorte ces réponses mises en antémémoire pour d'autres résolveurs, réduisant considérablement par là l'effet de la mise en antémémoire. Le présent document traite les questions soulevées à la lumière de l'expérience et remplace le paragraphe 4.3.4 de la [RFC1034].

La mise en antémémoire négative était une partie facultative de la spécification du DNS et traite de la mise en antémémoire de la non existence d'un RRset [RFC2181] ou nom de domaine.

La mise en antémémoire négative est utile car elle réduit le temps de réponse pour les réponses négatives. Elle réduit aussi le nombre de messages qui doivent être envoyés entre les résolveurs et les serveurs de noms, et donc le trafic réseau global. Une large proportion du trafic du DNS sur l'Internet pourrait être éliminée si tous les résolveurs mettaient en œuvre la mise en antémémoire négative. Avec cet objectif, la mise en antémémoire négative ne devrait plus être vue comme une partie facultative d'un résolveur du DNS.

Table des matières

1. Terminologie

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" dans ce document sont à interpréter comme décrit dans la [RFC2119].

"Mise en antémémoire négative" – C'est la mémorisation de la connaissance que quelque chose n'existe pas. On peut mémoriser la connaissance qu'un enregistrement a une valeur particulière. On peut aussi faire l'inverse, c'est-

à-dire, mémoriser la connaissance qu'un enregistrement n'existe pas. La mémorisation de la connaissance que quelque chose n'existe pas ou ne peut pas faire quelque chose donne une réponse qu'on appelle mise en antémémoire négative.

"QNAME" – C'est le nom d'une réponse dans la section d'interrogation, ou lorsque cela se résout en un CNAME, ou en une chaîne de CNAME, le champ de données du dernier CNAME. Le dernier CNAME dans ce sens est celui qui contient une valeur qui ne se résout pas en un autre CNAME. Les mises en œuvre devraient noter qu'inclure des enregistrements CNAME dans les réponses dans l'ordre, de sorte que la première ait l'étiquette de la section d'interrogation, puis que chacune à la suite ait l'étiquette de la section de données de la précédente (cela nécessite plus d'un CNAME) permet à la séquence d'être traitée en une seule passe, et facilite considérablement la tâche du receveur. D'autres enregistrements pertinents (comme les RR SIG [RFC2065]) peuvent être insérés parmi les CNAME.

"NXDOMAIN" – C'est une expression de remplacement pour le RCODE "Erreur de nom" décrite au paragraphe 4.1.1 de la [RFC1035] et les deux termes sont utilisés de façon interchangeable dans le présent document.

"NODATA" – C'est un pseudo RCODE qui indique que le nom est valide, pour une certaine classe, mais qu'il n'y a pas d'enregistrement de ce type. Une réponse NODATA doit être déduite de la réponse.

"FORWARDER" – C'est un serveur de noms utilisé pour résoudre des interrogations au lieu d'utiliser directement la chaîne de serveurs de noms d'autorité. Le transmetteur a normalement un meilleur accès à l'Internet, ou entretient une plus grosse antémémoire qui peut être partagée par de nombreux résolveurs. Il sort du domaine d'application du présent document de déterminer comment un serveur est identifié comme FORWARDER, ou sait qu'il est un FORWARDER. Cependant, si vous êtes utilisé comme transmetteur, l'interrogation aura le fanion Récurrence désirée établi.

La lecture du présent document suppose pour sa compréhension celle des [RFC1034], [RFC1035] et [RFC2065].

2. Réponses négatives

Les réponses négatives les plus courantes indiquent qu'un RRset particulier n'existe pas dans le DNS. Les premières sections du présent document traitent de ce cas. Les autres réponses négatives peuvent indiquer des défaillances d'un serveur de noms, et elles sont traitées à la section 7 (Autres réponses négatives).

Une réponse négative est indiquée par une des conditions suivantes :

2.1 Erreur de nom

Les erreurs de nom (NXDOMAIN) sont indiquées par la présence de "Erreur de nom" dans le champ RCODE. Dans ce cas, le domaine auquel se réfère le QNAME n'existe pas. Noter que la section réponse peut avoir des RR SIG et CNAME et que la section autorité peut avoir des RR SOA, NXT [RFC2065] et SIG.

Il est possible de distinguer un référant et une réponse NXDOMAIN par la présence de NXDOMAIN dans le RCODE sans considération de la présence d'enregistrements NS ou SOA dans la section autorité.

Les réponses NXDOMAIN peuvent être rangées en quatre types par le contenu de la section autorité. Ils sont indiqués ci-dessous ainsi qu'avec un référant pour comparaison. Les champs qui ne sont pas mentionnés ne sont pas importants selon les termes des exemples.

RÉPONSE NXDOMAIN : TYPE 1.
En-tête : RDCODE=NXDOMAIN
Interrogation : AN.EXAMPLE. A
Réponse : AN.EXAMPLE. CNAME TRIPPLE.XX.
Autorité : XX. SOA NS1.XX. HOSTMASTER.NS1.XX. ....
XX. NS NS1.XX.
XX. NS NS2.XX.
Supplément :
NS1.XX. A 127.0.0.2
NS2.XX. A 127.0.0.3

RÉPONSE NXDOMAIN : TYPE 2.
En-tête : RDCODE=NXDOMAIN
Interrogation : AN.EXAMPLE. A
Réponse : AN.EXAMPLE. CNAME TRIPPLE.XX.
Autorité : XX. SOA NS1.XX. HOSTMASTER.NS1.XX. ....
Supplément : <vide>

RÉPONSE NXDOMAIN : TYPE 3.
En-tête : RDCODE=NXDOMAIN
Interrogation : AN.EXAMPLE. A
Réponse : AN.EXAMPLE. CNAME TRIPPLE.XX.
Autorité : <vide>
Supplément : <vide>

RÉPONSE NXDOMAIN : TYPE 4.
En-tête : RDCODE=NXDOMAIN
Interrogation : AN.EXAMPLE. A
Réponse : AN.EXAMPLE. CNAME TRIPPLE.XX.
Autorité : XX. NS NS1.XX.
XX. NS NS2.XX.
Supplément : NS1.XX. A 127.0.0.2
NS2.XX. A 127.0.0.3

RÉPONSE DE RÉFÉRANT.
En-tête : RDCODE=NOERROR
Interrogation : AN.EXAMPLE. A
Réponse : AN.EXAMPLE. CNAME TRIPPLE.XX.
Autorité : XX. NS NS1.XX.
XX. NS NS2.XX.
Supplément : NS1.XX. A 127.0.0.2
NS2.XX. A 127.0.0.3

Noter que dans les quatre exemples de réponses NXDOMAIN, on sait que le nom "AN.EXAMPLE." existe, et a comme valeur un enregistrement CNAME. NXDOMAIN se réfère à "TRIPPLE.XX", dont on sait alors qu'il n'existe pas. D'un autre côté, dans l'exemple du référant, on montre que "AN.EXAMPLE" existe, et a un RR CNAME comme valeur, mais on ne sait rien du tout sur l'existence de "TRIPPLE.XX", en dehors de ce que "NS1.XX" ou "NS2.XX" peuvent être consultés en prochaine étape pour obtenir des informations sur lui.

Lorsque aucun enregistrement CNAME n'apparaît, la réponse NXDOMAIN se réfère au nom dans l'étiquette du RR dans la section question .

2.1.1 Traitement particulier de l'erreur de nom

Ce paragraphe traite des erreurs rencontrées lors de la mise en œuvre de la mise en antémémoire négative des réponses NXDOMAIN.

Il existe actuellement un grand nombre de résolveurs qui échouent à détecter correctement et traiter toutes les formes de réponse NXDOMAIN. Certains résolveurs traitent une réponse NXDOMAIN de TYPE 1 comme un référant. Pour atténuer ce problème, il est recommandé que les serveurs qui sont d'autorité pour la réponse NXDOMAIN n'envoient que des réponses NXDOMAIN de TYPE 2, c'est à dire que la section autorité contient un enregistrement SOA et pas d'enregistrement NS. Si un serveur qui n'est pas d'autorité envoie une réponse NXDOMAIN de type 1 à un de ces vieux résolveurs, le résultat sera une interrogation inutile à un serveur d'autorité. Ceci n'est pas souhaitable, mais pas fatal, sauf quand le serveur est utilisé comme Transmetteur. Cependant, si le résolveur utilise le serveur comme un transmetteur pour un tel résolveur, il sera nécessaire de désactiver l'envoi de sa réponse NXDOMAIN de TYPE 1 et d'utiliser à la place le NXDOMAIN de TYPE 2.

Certains résolveurs continuent de procéder de façon incorrecte si le fanion de réponse d'autorité n'est pas établi, tournant en boucle jusqu'à ce que le seuil de répétition d'interrogation soit dépassé et retournant alors à SERVFAIL. C'est un problème lorsque votre serveur de noms figure sur la liste des transmetteurs vers de tels résolveurs. Si le serveur de noms est utilisé comme transmetteur par un tel résolveur, le fanion d'autorité devra être établi de force sur les réponses NXDOMAIN à ces résolveurs. En pratique, cela ne cause pas de problème même si il n'est jamais établi et a le comportement par défaut dans BIND à partir de 4.9.3.

2.2 Pas de données

NODATA est indiqué par une réponse avec le RCODE réglé à NOERROR et pas de réponse pertinente dans la section réponse. La section autorité va contenir un enregistrement SOA, ou il n'y aura pas d'enregistrement NS.

Les réponses NODATA doivent être déterminées par un algorithme à partir du contenu de la réponse car il n'y a pas de valeur de RCODE pour indiquer NODATA.
Dans certains cas, pour déterminer avec certitude que NODATA est la réponse correcte, il peut être nécessaire d'envoyer une autre interrogation.

La section autorité peut contenir des RRset NXT et SIG en plus des enregistrements NS et SOA. Les enregistrements CNAME et SIG peuvent exister dans la section réponse.

Il est possible de faire une distinction entre une réponse NODATA et un référant par la présence d'un enregistrement SOA dans la section autorité ou par l'absence d'enregistrement NS dans la section autorité.

On peut faire entrer les réponses NODATA dans trois catégories de types par le contenu de la section autorité. Ces types sont indiqués ci-dessous avec un référant pour la comparaison. Les champs non mentionnés n'ont pas d'importance selon les termes de ces exemples.

RÉPONSE NODATA : TYPE 1.
En-tête : RDCODE=NOERROR
Interrogation : ANOTHER.EXAMPLE. A
Réponse : <vide>
Autorité : EXAMPLE. SOA NS1.XX. HOSTMASTER.NS1.XX. ....
EXAMPLE. NS NS1.XX.
EXAMPLE. NS NS2.XX.
Supplément : NS1.XX. A 127.0.0.2
NS2.XX. A 127.0.0.3

RÉPONSE NODATA : TYPE 2.
En-tête : RDCODE=NOERROR
Interrogation : ANOTHER.EXAMPLE. A
Réponse : <vide>
Autorité : EXAMPLE. SOA NS1.XX. HOSTMASTER.NS1.XX. ....
Supplément : <vide>

RÉPONSE NODATA : TYPE 3.
En-tête : RDCODE=NOERROR
Interrogation : ANOTHER.EXAMPLE. A
Réponse : <vide>
Autorité : <vide>
Supplément : <vide>

RÉPONSE DE RÉFÉRANT.
En-tête : RDCODE=NOERROR
Interrogation : ANOTHER.EXAMPLE. A
Réponse : <vide>
Autorité : EXAMPLE. NS NS1.XX.
EXAMPLE. NS NS2.XX.
Supplément : NS1.XX. A 127.0.0.2
NS2.XX. A 127.0.0.3

Ces exemples, à la différence des exemples NXDOMAIN ci-dessus, n'ont pas d'enregistrement CNAME, cependant, ils le pourraient, juste de la même façon qu'en avaient les exemples NXDOMAIN, auquel cas ce serait la valeur du dernier CNAME (le QNAME) pour lequel NODATA serait la conclusion.

2.2.1 Traitement particulier de Pas de données

Il existe actuellement un grand nombre de résolveurs qui ne réussissent pas à détecter correctement et à traiter toutes les formes de réponses NODATA. Certains résolveurs traitent une réponse NODATA de TYPE 1 comme un référant. Pour atténuer ce problème, il est recommandé que les serveurs qui sont d'autorité pour la réponse NODATA n'envoient que des réponses NODATA de TYPE 2, c'est à dire que la section autorité contient un enregistrement SOA et pas d'enregistrement NS. L'envoi d'une réponse NODATA de TYPE 1 à partir d'un serveur qui n'est pas d'autorité à un de ces résolveurs va avoir pour seul résultat une interrogation inutile. Si un serveur figure sur la liste comme transmetteur pour un autre résolveur, il peut aussi être nécessaire de désactiver l'envoi de réponses NODATA de TYPE 1 pour les réponses NODATA qui ne sont pas d'autorité.

Certains serveurs de noms ne réussissent pas à établir le RCODE à NXDOMAIN en présence de CNAME dans la section réponse. Si une réponse NXDOMAIN / NODATA définitive est requise dans ce cas, le résolveur doit interroger à nouveau en utilisant le QNAME comme étiquette de l'interrogation.

3. Réponses négatives de la part des serveurs d'autorité

Les serveurs de noms qui sont d'autorité pour une zone DOIVENT inclure l'enregistrement SOA de la zone dans la section autorité de la réponse lorsque ils font rapport d'un NXDOMAIN ou qu'ils indiquent qu'aucune donnée du type demandé n'existe. Ceci est exigé afin que la réponse puisse être mise en antémémoire. La TTL *(durée de vie)* de cet enregistrement est réglée au minimum du champ MINIMUM de l'enregistrement SOA et de la TTL du SOA lui-même, et indique pendant combien de temps un résolveur peut conserver en antémémoire la réponse négative. L'enregistrement SIG de la TTL associée à l'enregistrement SOA devrait aussi être aligné sur la TTL du SOA.

Si la zone contenante est signée selon la [RFC2065] le SOA et les enregistrements NXT et SIG appropriés DOIVENT être ajoutés.

4. Champ SOA Minimum

Le champ SOA Minimum a été surchargé dans le passé pour avoir trois significations différentes, la valeur minimum de TTL de tous les RR dans une zone, la TTL par défaut des RR qui ne contiennent pas de valeur de TTL, et la TTL des réponses négatives.

Bien qu'elle soit la signification définie à l'origine, la première d'entre elles, la valeur minimum de TTL de tous les RR dans une zone, n'a jamais été utilisée en pratique et est donc de ce fait déconseillée.
La seconde, la TTL par défaut des RR qui ne contiennent pas de TTL explicite dans le fichier de zone maître, n'est pertinent que sur le serveur principal. Après un transfert de zone, tous les RR ont une TTL explicite et il est impossible de déterminer si la TTL pour un enregistrement a été établie explicitement ou déduite de la valeur par défaut après un transfert de zone. Lorsque un serveur n'exige pas que les RR incluent explicitement la valeur de TTL, il devrait fournir un mécanisme, qui ne serait pas la valeur du champ MINIMUM de l'enregistrement SOA, à partir duquel les valeurs de TTL manquantes seront obtenues. La réalisation de ce mécanisme dépend de la mise en œuvre.

Le format du fichier maître de la section 5 de la [RFC1035] est étendu de façon à inclure la directive suivante :

$TTL <TTL> [commentaire]

Tous les enregistrements de ressource qui apparaissent après la directive, et qui n'incluent pas explicitement une valeur de TTL, ont leur TTL réglée à la TTL donnée dans la directive $TTL. Les enregistrements SIG sans TTL explicite obtiennent leur TTL de la "TTL d'origine" de l'enregistrement SIG du paragraphe 4.5 de la [RFC2065].

La dernière des significations actuelles, celle d'être la TTL utilisée pour les réponses négatives, est la nouvelle signification définie pour le champ SOA Minimum.

5. Mise en antémémoire de réponses négatives

Comme les réponses normales, les réponses négatives ont une durée de vie (TTL, *time to live*). Comme il n'y a pas d'enregistrement dans la section réponse auquel cette TTL puisse être appliquée, la TTL doit être portée par une autre méthode. Cela est fait par l'inclusion de l'enregistrement SOA provenant de la zone dans la section autorité de la réponse.

Lorsque le serveur d'autorité crée cet enregistrement, sa TTL est tirée du minimum du champ SOA.MINIMUM et de la TTL du SOA. Cette TTL se décrémente de façon similaire à celle d'une réponse normale mise en antémémoire et lorsque elle atteint zéro (0) indique que la réponse négative mise en antémémoire NE DOIT PAS être utilisée à nouveau.

Une réponse négative qui a résulté d'une erreur de nom (NXDOMAIN) devrait être mise en antémémoire de façon qu'elle puisse être restituée et retournée en réponse à une autre interrogation pour le même <QNAME, QCLASS> qui a résulté en la réponse négative mise en antémémoire.

Une réponse négative qui a résulté d'une erreur d'absence de données (NODATA) devrait être mise en antémémoire de telle sorte qu'elle puisse être restituée et retournée en réponse à une autre interrogation sur la même <QNAME, QTYPE, QCLASS> qui a résulté en la réponse négative mise en antémémoire.

L'enregistrement NXT, si il existe dans la section autorité d'une réponse négative reçue, DOIT être mémorisé de telle façon qu'elle puisse être localisée et retournée avec l'enregistrement SOA dans la section autorité, comme le serait tout enregistrement SIG dans la section autorité. Pour les réponses NXDOMAIN, il n'y a pas de relation "nécessaire" évidente entre les enregistrements NXT et le QNAME. L'enregistrement NXT DOIT avoir le même nom de propriétaire que le nom d'interrogation pour les réponses NODATA.

Les réponses négatives sans enregistrement SOA NE DEVRAIENT PAS être mises en antémémoire car il n'y a aucun moyen d'empêcher une boucle sans fin de réponses négatives entre une paire de serveurs, même avec une TTL courte.

Bien que le DNS forme une arborescence de serveurs, avec diverses erreurs de configuration, il est possible de former une boucle dans le graphe des interrogations, par exemple, deux serveurs s'inscrivant l'un l'autre comme transmetteur, diverses configurations de serveur boiteuses. Sans décompte de la TTL, une mise en antémémoire de réponse négative aurait sa TTL rétablie lorsque elle est reçue par le serveur suivant. Cette indication négative pourrait alors durer à jamais en circulant entre les serveurs concernés.

Comme avec les réponses positives de mise en antémémoire, il est important pour un résolveur de limiter la durée pendant laquelle il va garder en antémémoire une réponse négative car le protocole prend en charge la mise en antémémoire jusqu'à 68 ans. Une telle limite ne devrait pas être supérieure à celle appliquée aux réponses positives et de préférence elle devrait être réglable. Il a été trouvé que des valeurs de une à trois heures fonctionnent bien et feraient une valeur pas défaut convenable. Les valeurs qui dépassent un jour se sont révélées problématiques.

6. Réponses négatives de la part de l'antémémoire

Lorsque un serveur, en répondant à une interrogation, rencontre une réponse négative en antémémoire, il DOIT ajouter l'enregistrement SOA de l'antémémoire à la section autorité de la réponse avec le TTL décrémenté de la durée pendant laquelle elle a été conservée dans l'antémémoire. Cela permet à la réponse NXDOMAIN / NODATA de se périmer correctement.

Si un enregistrement NXT a été mis en antémémoire avec un enregistrement SOA, il DOIT être ajouté à la section autorité. Si un enregistrement SIG a été mis en antémémoire avec un enregistrement NXT, il DEVRAIT être ajouté à la section autorité.

Comme avec toutes les réponses provenant de l'antémémoire, les réponses négatives DEVRAIENT avoir un référant implicite incorporé dans la réponse. Ceci permet au résolveur de localiser une source d'autorité. Un référant implicite est caractérisé par des enregistrements NS dans la section autorité qui se réfèrent au résolveur à l'égard d'une source d'autorité. Les réponses NXDOMAIN des types 1 et 4 contiennent des référants implicites comme le font les réponses NODATA de type 1.

7. Autres réponses négatives

La mise en antémémoire des autres réponses négatives n'est pas couverte par une RFC existante. Il n'y a aucun moyen d'indiquer une TTL désirée dans ces réponses. Il faut veiller à s'assurer qu'il n'y a pas de boucles dans la transmission.

7.1 Défaillance du serveur (FACULTATIF)

Les défaillances de serveur entrent dans deux classes majeures. La première est lorsque un serveur peut déterminer qu'il a

été mal configuré pour une zone. Cela peut venir de ce qu'il a été noté comme étant un serveur, mais qu'il n'est pas configuré comme serveur pour la zone, ou de ce qu'il a été configuré comme serveur pour la zone, mais ne peut obtenir pour une raison quelconque les données de la zone. Cela peut arriver soit parce que le fichier de zone n'existe pas, soit qu'il contient des erreurs, ou parce qu'un autre serveur à partir duquel la zone devrait être disponible ne répond pas ou est incapable de fournir la zone ou ne veut pas la fournir.

La seconde classe est lorsque le serveur a besoin d'obtenir une réponse d'ailleurs, mais est incapable de la faire, du fait de défaillances du réseau, d'autres serveurs qui ne répondent pas, ou retournent des erreurs de défaillance de serveur, ou des causes similaires.

Dans l'un et l'autre cas, un résolveur PEUT mettre en antémémoire une réponse de défaillance d'un serveur. Si il le fait, il NE DOIT PAS la mettre en antémémoire pour plus de cinq (5) minutes, et elle DOIT être mise en antémémoire avec le tuplet spécifique d'interrogation <nom d'interrogation, type, classe, adresse IP du serveur>.

7.2 Serveur mort/injoignable (FACULTATIF)

Les serveurs morts/injoignables sont des serveurs qui échouent à répondre d'une façon ou d'une autre à une interrogation ou lorsque la couche transport a fourni une indication que le serveur n'existe pas ou est injoignable. Un serveur peut être réputé mort ou injoignable si il n'a pas répondu à une interrogation en cours dans les 120 secondes.

Des exemples d'indications de la couche transport sont des messages d'erreur ICMP qui indiquent un hôte, réseau ou accès injoignable. TCP rétablit les messages d'erreur de la pile IP en fournissant des indications similaires à celles ci-dessus.

Un résolveur PEUT mettre en antémémoire une indication de serveur mort. Si il le fait, il NE DOIT PAS être réputé mort pour plus de cinq (5) minutes. L'indication DOIT être mémorisée en regard du tuplet d'interrogation <nom de l'interrogation, type, classe, adresse IP du serveur> sauf si il y avait une indication de la couche transport disant que le serveur n'existe pas, auquel cas elle s'applique à toutes les interrogations à cette adresse IP spécifique.

8. Changements depuis la RFC1034

La mise en antémémoire négative dans les résolveurs n'est plus facultative, si un résolveur met en antémémoire quelque chose, il doit aussi mettre en antémémoire les réponses négatives.

Les réponses négatives qui ne sont pas d'autorité PEUVENT être mises en antémémoire.

L'enregistrement SOA provenant de la section autorité DOIT être mis en antémémoire. Les indications d'erreur de nom doivent être mises en antémémoire en regard du tuplet <nom d'interrogation, QCLASS>. Aucune indications de données ne doit être mise en antémémoire avec le tuplet <nom d'interrogation, QTYPE, QCLASS>.

Un enregistrement SOA en antémémoire doit être ajouté à la réponse. Ceci était explicitement interdit parce que précédemment la distinction entre un enregistrement SOA en antémémoire normal et le SOA mis en antémémoire par suite d'une réponse négative n'était pas faite, et extraire simplement un SOA normal en antémémoire et l'ajouter à une réponse négative en antémémoire cause des problèmes.

La directive TTL $TTL a été ajouté au format de fichier maître.

9. Historique de la mise en antémémoire négative

La présente section présente un bref historique le la mise en antémémoire négative dans le DNS et ne fait pas partie de la spécification technique de la mise en antémémoire négative.

Il est intéressant de noter que les mêmes concepts ont été réinventé à la fois dans les serveurs CHIVES et BIND.

L'histoire des premiers travaux CHIVES (paragraphe 9.1) a été fournie par Rob Austein <sra@epilogue.com> et est reproduite ici dans la forme sous laquelle il l'a fournie [MPA].

Dans le courant du printemps 1985, j'ai mentionné à Paul Mockapetris que notre expérience de son résolveur DNS JEEVES avait démontré le besoin d'une forme de schéma de mise en antémémoire négative. Paul suggéra qu'on mette simplement en

antémémoire les erreurs d'autorité, en utilisant la valeur SOA MINIMUM pour la zone qui contiendrait les enregistrements de ressource cibles. Je suis tout à fait certain que cette conversation a eu lieu avant la rédaction de la RFC-973, mais je n'ai jamais su précisément si cette idée était venue spontanément à Paul en réponse à ma question ou si c'est quelque chose qu'il prévoyait de mettre dans le document qui est devenu la RFC-973. Dans tous les cas, aucun de nous n'était entièrement sûr que la valeur de SOA MINIMUM était réellement la bonne métrique à utiliser, mais elle était disponible et était sous le contrôle de l'administrateur de la zone cible, et les deux nous semblaient à l'époque des caractéristiques importantes.

À la fin 1987, j'ai publié la version initiale des beta-test de CHIVES, le résolveur DNS, que j'avais écrite pour remplacer le résolveur JEEVES de Paul. CHIVES comportait un mécanisme de recherche de chemin qui était utilisé de façon très intense sur plusieurs sites (y compris le mien) de sorte que CHIVES comportait aussi un mécanisme de mise en antémémoire négative fondé sur les valeurs MINIMUM de SOA. La stratégie de base était de mettre en antémémoire les codes d'erreur d'autorité classés par les paramètres exacts d'interrogation (QNAME, QCLASS, et QTYPE) avec une TTL d'antémémoire égale à la valeur MINIMUM du SOA. CHIVES ne cherchait pas à suivre à la trace les RR SOA pour savoir si ils n'étaient pas fournis dans la réponse d'autorité, de sorte qu'il ne s'est jamais soucier d'éliminer complètement le trafic gratuit de messages d'erreur du DNS, mais cela apportait une aide considérable. Il faut se souvenir que cela ce passait a peu près au même moment que le presque effondrement de l'ARPANET dû à l'encombrement causé par la croissance exponentielle et le "vieil" (pré-VJ) algorithme de retransmission TCP, si bien que la mise en antémémoire négative résultait en des temps de réponse du DNS drastiquement améliorés pour nos usagers, pour les démons de messagerie, et cœtera.

Pour autant que je le sache, CHIVES était le premier résolveur à mettre en œuvre la mise en antémémoire négative. CHIVES a été développé durant les années du crépuscule de TOPS-20, de sorte qu'il n'a jamais fonctionné sur un grand nombre de machines, mais les quelques unes sur lesquelles il fonctionnait étaient celles qu'il fallait coûte que coûte fermer rapidement. Ainsi les quelques utilisateurs que nous avions tendaient à piloter CHIVES sans douceur. Plusieurs questions intéressantes de la technologie du DNS ont résulté de cela, mais celle qui nous intéresse ici est le paramètre de configuration MAXTTL.

L'expérience avec JEEVES avait déjà montré que les RR affichaient souvent des TTL ridiculement longues (99999999 était particulièrement populaire pendant de nombreuses années, du fait de bogues dans le code et dans la documentation de plusieurs versions précoces de BIND) et le logiciel robuste qui croyait aveuglément de telles TTL pouvaient créer tellement de défaillances étranges qu'il était souvent nécessaire de réamorcer fréquemment le résolveur juste pour nettoyer ce fatras de l'antémémoire. Ainsi CHIVES avait un paramètre de configuration "MAXTTL", qui spécifiait la TTL maximum "raisonnable" dans un RR reçu. Les RR avec des TTL supérieures à MAXTTL auraient leur TTL réduite à MAXTTL ou seraient entièrement éliminés, selon le réglage d'un autre paramètre de configuration.

Quand nous avons commencé à avoir l'expérience du terrain avec le code de mise en antémémoire négative de CHIVES, il est devenu clair que la valeur MINIMUM de SOA était souvent assez grande pour causer le même type de problèmes pour la mise en antémémoire négative que les énormes TTL dans les RR en avaient pour la mise en antémémoire normale (là encore, ceci était en partie dû à une bogue dans plusieurs versions précoces de BIND là où un serveur secondaire nierait d'autorité toute connaissance de ses zones si il ne pouvait pas contacter les principaux au réamorçage). Ainsi, nous avons commencé à faire fonctionner aussi les TTL d'antémémoire négative à travers la vérification de MAXTTL, et nous avons continué d'expérimenter.

La configuration qui semblait fonctionner le mieux sur WSMR-SIMTEL20.ARMY.MIL (la dernière des machines TOPS-20 majeures de l'Internet à être fermée, donc le dernier utilisateur majeur de CHIVES, donc l'endroit qui nous fournissait la plus longue plate-forme expérimentale) était de régler MAXTTL à environ trois jours. La plus grande partie du trafic généré par SIMTEL20 dans ses dernières années était en rapport avec la messagerie et le temporisateur de file d'attende de la messagerie était réglé à une semaine, de sorte que cela donnait à un message "collé" plusieurs essais d'achèvement de résolution DNS, sans couler le système avec un flot d'interrogations inutiles. Comme (pour des raisons qui m'échappent aujourd'hui) nous n'avions que le seul paramètre MAXTTL plutôt que des paramètres distincts pour la mise en antémémoire positive et négative, on ne sait pas précisément quel effet avait ce réglage de MAXTTL sur le code de mise en antémémoire négative.

CHIVES incluait aussi un second mécanisme, quelque peu controversé, qui tenait lieu dans certains cas de mise en antémémoire négative. Le démon de résolveur CHIVES pouvait être configuré de façon à charger des fichiers maîtres du DNS, ce qui lui donnait la capacité d'agir comme ce qu'on appellerait aujourd'hui un "secondaire fugitif". C'est-à-dire que configuré de cette façon, le résolveur avait un accès direct aux informations d'autorité pour les zones de forte utilisation. Les mécanismes de recherche de chemin de CHIVES reflétaient cela : il y avait en fait deux chemins de recherche séparés, un pour la seule recherche des données de zone locale d'autorité, et un qui pouvait générer des interrogations itératives normales. Ce découpage du besoin de mise en antémémoire négative dans les cas où on pouvait prévoir une utilisation lourde (par exemple, le résolveur sur XX.LCS.MIT.EDU chargeait toujours les fichiers de zone pour LCS.MIT.EDU et pour AI.MIT.EDU et mettait ces deux suffixes dans le chemin de recherche "local" car entre eux, les hôtes de ces deux zones constituaient le gros du trafic DNS). Tous les sites qui fonctionnaient avec CHIVES ne choisissaient pas d'utiliser cette caractéristique ; C.CS.CMU.EDU, par exemple, avait choisi d'utiliser le chemin de recherche "distant" pour tout parce

qu'il y avait trop de sous-zones différentes au CMU pour que l'ombrage de zone soit praticable, de sorte qu'il s'appuyait très fortement sur la mise en antémémoire négative même pour le trafic local.

Globalement, je pense tout de même que le concept de base que nous avons utilisé pour la mise en antémémoire négative était très raisonnable : l'administrateur de zone spécifiait pendant combien de temps mettre en antémémoire les réponses négatives, et la configuration du résolveur choisissait la durée réelle de mise en antémémoire dans la gamme entre zéro et la période spécifiée par l'administrateur de zone. Il y a un certain nombre de détails que je traiterais différemment aujourd'hui (comme d'utiliser un nouveau champ SOA au lieu de surcharger le champ MINIMUM) mais après plus de dix ans, j'aurais été surpris qu'on ne trouve pas au moins quelques améliorations.

9.2 BIND

Bien que ce ne soit pas la première tentative d'obtenir une mise en antémémoire négative dans BIND, en juillet 1993, Anant Kumar de l'ISI fournit au BIND 4.9.2 ALPHA le code qui mettait en œuvre la validation et la mise en antémémoire négative (NCACHE). Ce code avait une TTL de 10 minutes pour la mise en antémémoire négative et ne mettait en antémémoire que l'indication qu'il y avait une réponse négative, NXDOMAIN ou NOERROR_NODATA. C'est l'origine du code de pseudo-réponse NODATA mentionné plus haut.

Mark Andrews de CSIRO a ajouté du code (RETURNSOA) qui mémorisait l'enregistrement SOA de telle sorte qu'il puisse être restitué par une interrogation similaire. UUnet s'est plaint de ce qu'ils obtenaient des vieilles réponses après avoir téléchargé une nouvelle zone, et que l'option était désactivée, sur BIND 4.9.3-alpha5, en avril 1994. En réalité, cela indiquait que le serveur avait besoin de purger l'espace occupé par la zone. La fonctionnalité pour le faire a été ajoutée dans BIND 4.9.3 BETA11 patch2 en décembre 1994.

RETURNSOA a été réactivé par défaut, dans BIND 4.9.5-T1A, en août 1996.

10. Exemple

L'exemple suivant se fonde sur une zone signée qui est vide à part les serveurs de noms. On va interroger WWW.XX.EXAMPLE et montrer la réponse initiale et de nouveau 10 minutes plus tard.
Note 1 : durant ces 10 minutes, les enregistrements NS pour XX.EXAMPLE sont arrivés à expiration.
Note 2 : la TTL de l'enregistrement SIG n'est pas établie de façon explicite dans le fichier zone et est donc la TTL du RRset dont il est la signature.

Fichier zone :

```
$TTL 86400
$ORIGIN XX.EXAMPLE.
@       IN      SOA     NS1.XX.EXAMPLE. HOSTMATER.XX.EXAMPLE. (
                                1997102000      ; serial
                                1800            ; refresh (30 m)
                                900             ; retry (15 m)
                                604800          ; expire (7 jours)
                                1200 )          ; minimum (20 m)
        IN      SIG     SOA ...
  1200  IN      NXT     NS1.XX.EXAMPLE. A NXT SIG SOA NS KEY
        IN      SIG     NXT ... XX.EXAMPLE. ...
   300  IN      NS      NS1.XX.EXAMPLE.
   300  IN      NS      NS2.XX.EXAMPLE.
        IN      SIG     NS ... XX.EXAMPLE. ...
        IN      KEY     0x4100 1 1 ...
        IN      SIG     KEY ... XX.EXAMPLE. ...
        IN      SIG     KEY ... EXAMPLE. ...
NS1     IN      A       10.0.0.1
        IN      SIG     A ... XX.EXAMPLE. ...
  1200  IN      NXT     NS2.XX.EXAMPLE. A NXT SIG
        IN      SIG     NXT ...
NS2     IN      A       10.0.0.2        IN  SIG     A ... XX.EXAMPLE. ...
  1200  IN      NXT     XX.EXAMPLE. A NXT SIG
        IN      SIG     NXT ... XX.EXAMPLE. ...
```

```
Réponse initiale :
En-tête :       RDCODE=NXDOMAIN, AA=1, QR=1, TC=0
Interrogation : WWW.XX.EXAMPLE. IN A
Réponse :       <vide>
Autorité :      XX.EXAMPLE.      1200 IN SOA NS1.XX.EXAMPLE. ...
                XX.EXAMPLE.      1200 IN SIG SOA ... XX.EXAMPLE. ...
                NS2.XX.EXAMPLE.  1200 IN NXT XX.EXAMPLE. NXT A NXT SIG
                NS2.XX.EXAMPLE.  1200 IN SIG NXT ... XX.EXAMPLE. ...
                XX.EXAMPLE.     86400 IN NS  NS1.XX.EXAMPLE.
                XX.EXAMPLE.     86400 IN NS  NS2.XX.EXAMPLE.
                XX.EXAMPLE.     86400 IN SIG NS ... XX.EXAMPLE. ...
Supplément :    XX.EXAMPLE.     86400 IN KEY 0x4100 1 1 ...
                XX.EXAMPLE.     86400 IN SIG KEY ... EXAMPLE. ...
                NS1.XX.EXAMPLE. 86400 IN A   10.0.0.1
                NS1.XX.EXAMPLE. 86400 IN SIG A ... XX.EXAMPLE. ...
                NS2.XX.EXAMPLE. 86400 IN A   10.0.0.2   NS3.XX.EXAMPLE. 86400 IN SIG A ... XX.EXAMPLE. ...

Après 10 minutes :
En-tête :       RDCODE=NXDOMAIN, AA=0, QR=1, TC=0
Interrogation : WWW.XX.EXAMPLE. IN A
Réponse :       <vide>
Autorité:       XX.EXAMPLE.      600 IN SOA NS1.XX.EXAMPLE. ...
                XX.EXAMPLE.      600 IN SIG SOA ... XX.EXAMPLE. ...
                NS2.XX.EXAMPLE.  600 IN NXT XX.EXAMPLE. NXT A NXT SIG
                NS2.XX.EXAMPLE.  600 IN SIG NXT ... XX.EXAMPLE. ...
                EXAMPLE.       65799 IN NS  NS1.YY.EXAMPLE.
                EXAMPLE.       65799 IN NS  NS2.YY.EXAMPLE.
                EXAMPLE.       65799 IN SIG NS ... XX.EXAMPLE. ...
Supplément :    XX.EXAMPLE.    65800 IN KEY 0x4100 1 1 ...
                XX.EXAMPLE.    65800 IN SIG KEY ... EXAMPLE. ...
                NS1.YY.EXAMPLE. 65799 IN A   10.100.0.1
                NS1.YY.EXAMPLE. 65799 IN SIG A ... EXAMPLE. ...
                NS2.YY.EXAMPLE. 65799 IN A   10.100.0.2
                NS3.YY.EXAMPLE. 65799 IN SIG A ... EXAMPLE. ...
                EXAMPLE.       65799 IN KEY 0x4100 1 1 ...
                EXAMPLE.       65799 IN SIG KEY ... . ...
```

11. Considérations pour la sécurité

On estime que le présent document n'introduit pas de menaces supplémentaires significatives pour la sécurité autres que celles qui existent déjà lorsque on utilise des données provenant du DNS.

Avec la mise en antémémoire négative, il serait possible de propager une attaque de déni de service en répandant un message NXDOMAIN avec une TTL très élevée. Sans la mise en antémémoire négative, cela serait beaucoup plus difficile. Un effet similaire aurait été obtenu précédemment en répandant un mauvais enregistrement A, afin que le serveur ne puisse pas être atteint – ce qui est presque la même chose. Cela a le même effet pour autant que l'utilisateur final soit capable de le faire mais avec un effet psychologique différent. Avec le mauvais A, je pense "Bon sang, le réseau est encore en panne" et je recommencerai demain. Avec le "NXDOMAIN" je pense "Oh, ils ont coupé le serveur et il n'existe plus" et je ne vais probablement jamais me soucier d'essayer ce serveur une autre fois.

Un exemple pratique de cette situation est celle d'un serveur SMTP où ce comportement est codé. Avec une attaque NXDOMAIN, le message serait refusé immédiatement, alors qu'avec une attaque de mauvais A, le message serait mis en file d'attente et pourrait passer après la suspension de l'attaque.

Pour qu'une telle attaque réussisse, le placement de NXDOMAIN doit être injecté dans un serveur parent (ou un résolveur d'antémémoire occupé). Une façon de faire cela serait d'utiliser un CNAME d'où il résulterait que le serveur parent interrogerait le serveur d'un attaquant. Les résolveurs qui souhaitent se prémunir contre de telles attaques peuvent interroger à nouveau le QNAME final en ignorant toutes les données NS dans les réponses qu'il a reçues pour cette interrogation.

La mise en œuvre de vérification de bonne santé des TTL va réduire l'efficacité de telles attaques, parce que la réussite de

l'attaque exigerait la réinjection des données pirates à des intervalles plus fréquents.

La sécurité du DNS de la [RFC2065] donne un mécanisme pour vérifier si une réponse négative est valide ou non, par l'utilisation d'enregistrements NXT et SIG. Le présent document soutient l'utilisation de ce mécanisme en recommandant la transmission des enregistrements de sécurité pertinents même dans un serveur sans capacité de sécurité.

Remerciements

Je tiens à remercier Rob Austein pour cet historique du serveur de noms CHIVES, ainsi que le groupe de travail DNSIND, et en particulier Robert Elz pour ses précieuses contributions techniques et rédactionnelles au présent document.

Références

[RFC1034] P. Mockapetris, "Noms de domaines - Concepts et facilités", STD 13, novembre 1987.

[RFC1035] P. Mockapetris, "Noms de domaines – Mise en œuvre et spécification", STD 13, novembre 1987.

[RFC2065] D. Eastlake 3rd, C. Kaufman, "Extensions de sécurité du système de noms de domaines", janvier 1997. (*Obsolète, voir* RFC2535) *(P.S.)*

[RFC2119] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997.

[RFC2181] R. Elz et R. Bush, "Clarifications pour la spécification du DNS", juillet 1997. *(Information)*

Adresse de l'auteur

Mark Andrews
CSIRO - Mathematical and Information Sciences
Locked Bag 17
North Ryde NSW 2113
AUSTRALIA
téléphone : +61 2 9325 3148
mél : Mark.Andrews@cmis.csiro.au

Déclaration complète de droits de reproduction