Groupe de travail Réseau
Request for Comments : 3958
Catégorie : En cours de normalisation
Traduction Claude Brière de L'Isle

L. Daigle, VeriSign, Inc.
A. Newton, VeriSign, Inc.

janvier 2005

Localisation de service d'application fondée sur le domaine avec les enregistrements de ressource de SRV et le service de recherche dynamique de délégation (DDDS)

Statut de ce mémoire

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

Résumé

Le présent mémoire définit un mécanisme généralisé pour la désignation de services d'application qui permet la localisation de services sans s'appuyer sur des conventions rigides de désignation de domaine (ce qu'on appelle les "name hack"). La proposition définit une application de système de découverte dynamique de délégation (DDDS, *Dynamic Delegation Discovery System*) pour transposer de façon dynamique le nom de domaine, le nom de service d'application, et le protocole d'application en serveur et accès cibles.

Table des Matières

1. Introduction

Le présent mémoire définit un mécanisme généralisé pour désigner un service d'application qui permet la localisation de service sans s'appuyer sur des conventions rigides de désignation de domaine (ce qu'on appelle des "name hacks" ou taxis de noms). La proposition définit une application de système de découverte dynamique de délégation (DDDS, *Dynamic Delegation Discovery System*) [RFC3401] pour transposer de façon dynamique le nom de domaine, le nom de service d'application et le protocole d'application en serveur et accès cibles.

Comme exposé à la Section 5, les approches existantes d'utilisation des enregistrements du DNS pour déterminer de façon dynamique l'hôte actuel d'un certain service d'application sont limitées en termes de cas d'utilisation pris en charge. Pour traiter certaines de ces limitations, le présent document définit une application DDDS pour transposer un "service + protocole + domaine" en adresses de serveur spécifiques par l'utilisation des enregistrements de ressource NAPTR [RFC3403] et SRV [RFC2782] du DNS. Ceci peut être vu comme une version plus générale de l'utilisation de SRV et/ou une application très restreinte de l'utilisation des enregistrements de ressource NAPTR.

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" en majuscules dans ce document sont à interpréter comme décrit dans le BCP 14, [RFC2119].

2. Spécification de NAPTR direct (S-NAPTR, Straightforward-NAPTR)

Les détails précis de la spécification de cette application DDDS sont donnés à la Section 6. Cette section définit l'usage de l'application DDDS.

2.1 Termes clés

"Service d'application" est un terme générique pour certains types d'application, indépendamment du protocole qui peut être utilisé pour l'offrir. Chaque service d'application sera associé à une étiquette enregistrée par l'IANA. Par exemple, la restitution de messagerie est un type de service d'application qui peut être mis en œuvre par différents protocoles de couche d'application (par exemple, POP3, IMAP4). Une étiquette comme "RetMail", pourrait être enregistrée pour cela. (Noter que ceci n'a pas été fait, et il n'est pas prévu de le faire au moment de cette rédaction.)

Les "protocoles d'application" sont utilisés pour mettre en œuvre des services d'application. Ils sont aussi associés à des étiquettes enregistrées par l'IANA. Utilisant l'exemple de la messagerie ci-dessus, "POP3" et "IMAP4" pourraient être enregistrés comme étiquettes de protocole d'application. Si plusieurs transports sont disponibles pour l'application, des étiquettes séparées devraient être définies pour chaque transport.

L'intention est que la combinaison des étiquettes de service d'application et de protocole devrait être assez spécifique pour que trouver une paire connue (par exemple) "RetMail:POP3" soit suffisant pour qu'un client identifie un serveur avec lequel il puisse communiquer.

Certains protocoles prennent en charge plusieurs services d'application. Par exemple, LDAP est un protocole d'application et peut se trouver prendre en charge divers services (par exemple, "pages blanches", "réseau à capacité de répertoire".

2.2 Usage de l'application de DDDS aux S-NAPTR

Comme défini à la Section 6, les enregistrements NAPTR sont utilisés pour mémoriser des informations de service d'application + protocole pour un certain domaine. Suivant la norme DDDS, ces enregistrements sont examinés, et les

règles de réécriture (contenues dans les enregistrements NAPTR) sont utilisées pour déterminer les recherches successives de DNS jusqu'à trouver une cible souhaitée.

Pour le reste de cette section, se reporter à l'ensemble d'enregistrements de ressource NAPTR pour exemple.com, montré dans la figure ci-dessous, où "WP" est l'étiquette de service d'application imaginée pour "white pages" et "EM" est l'étiquette de service d'application pour un service d'application imaginaire "Extensible Messaging".

```
exemple.com.
;;              ordr    préférenc   fanions
                e       e
IN NAPTR        100     10          ""      "WP:whois++"                    ( ; service
                                            "                                 ; regexp
                                            bunyip.exemple.                   ; remplacement
                                                                            )
IN NAPTR        100     20          "s"     "WP:ldap"                       (  ; service
                                            ""                                ; regexp
                                            _ldap._tcp.myldap.exemple.com.    ; remplacement
                                                                            )
IN NAPTR        200     10          ""      "EM:protA"                      (  ; service
                                            ""                                ; regexp
                                            sp.exemple.                       ; remplacement
                                                                            )
IN NAPTR        200     30          "a"     "EM:protB"                        ; service
                                            ""                                ; regexp
                                            myprotB.exemple.com.              ; remplacement
                                                                            )
```

2.2.1 Ordre et préférence

Un client récupère tous les enregistrements NAPTR associés au nom de domaine cible (exemple.com, ci-dessus). Il vont être triés en termes de ORDER croissant et de PREF croissante au sein de chaque ORDER.

2.2.2 Enregistrements NAPTR correspondants et non correspondants

En commençant par le premier enregistrement NAPTR trié, le client examine le champ SERVICE pour trouver une correspondance. Dans le cas de l'application DDDS S-NAPTR, cela signifie un champ SERVICE qui comporte les étiquettes pour le service d'application désiré et un protocole d'application pris en charge.

Si plus d'un enregistrement NAPTR correspond, ils sont traités dans l'ordre de tri croissant.

2.2.3 Enregistrements NAPTR terminaux et non terminaux

Un enregistrement NAPTR avec un champ FLAG vide est "non terminal" – c'est-à-dire, on doit effectuer plus d'une recherche de RR NAPTR. Donc, pour traiter un enregistrement NAPTR avec un champ FLAG vide dans S-NAPTR, le champ REPLACEMENT est utilisé comme cible de la prochaine recherche DNS -- pour des RR NAPTR.

Dans S-NAPTR, les seuls fanions terminaux sont "S" et "A". On les appelle des "terminaux" de recherche NAPTR parce qu'ils notent la fin des règles de traitement de DDDS/NAPTR. Dans le cas d'un fanion "S", le champ REPLACEMENT est utilisé comme cible d'une interrogation du DNS sur les RR SRV, et le traitement normal de SRV est appliqué. Dans le cas d'un fanion "A", un enregistrement d'adresse est recherché pour le champ REPLACEMENT cible (et on suppose l'accès de protocole par défaut).

2.2.4 S-NAPTR et résolutions successives

Comme montré dans l'exemple ci-dessus, il est possible d'avoir plusieurs cibles éventuelles pour une seule paire service d'application + protocole. Elles doivent être suivies dans l'ordre jusqu'à ce qu'on réussisse à contacter un serveur ou que tous les enregistrements NAPTR correspondants possibles aient été poursuivis successivement par la recherche terminale et le contact de serveur. C'est-à-dire qu'un client doit faire marche arrière et tenter d'autres chemins de résolution en cas d'échec.

Un "échec" est déclaré, et la marche arrière doit être utilisée quand :

- o le serveur distant désigné (hôte et accès) échoue à fournir les accréditifs de sécurité appropriés pour le domaine *générateur* ;
- o la connexion avec le serveur distant désigné échoue pour une autre raison – les termes spécifiques en sont définis lorsque un protocole d'application est enregistré ; ou
- o la recherche DNS de S-NAPTR désigné échoue à donner les résultats attendus -- par exemple, pas de RR A pour une cible "A", pas d'enregistrement SRV pour une cible "S", ou pas d'enregistrement NAPTR avec un service d'application et protocole appropriés pour une recherche NAPTR. Sauf dans le cas de la toute première recherche NAPTR, cette dernière est une erreur de configuration : le fait que exemple.com ait un enregistrement NAPTR qui pointe sur "bunyip.exemple" pour le service et protocole "WP:Whois++" signifie que l'administrateur de exemple.com croit que ce service existe. Si bunyip.exemple n'a pas d'enregistrement NAPTR "WP:Whois++", le client d'application DOIT battre en retraite et essayer la prochaine option "WP:Whois++" disponible à partir de exemple.com. Comme il n'y en a pas, toute la résolution échoue.

Un client d'application interroge d'abord les RR NAPTR sur le domaine d'un certain service d'application. La première interrogation du DNS est pour les RR NAPTR dans le domaine cible d'origine (exemple.com, ci-dessus).

2.2.5 Clients qui prennent en charge plusieurs protocoles

Dans le cas d'un client d'application qui prend en charge plus d'un protocole pour un certain service d'application, il DOIT poursuivre complètement la résolution de S-NAPTR pour un protocole, en explorant toutes les recherches terminales potentielles selon les ordres de PREF et ORDER, jusqu'à ce que l'application réussisse à se connecter ou qu'il n'y ait plus de possibilités pour ce protocole.

C'est-à-dire que le client NE DOIT PAS commencer la recherche pour un protocole, observer qu'un ensemble de RR NAPTR successifs prend en charge un autre de ses protocoles préférés, et continuer la résolution de S-NAPTR sur la base de ce protocole. Par exemple, même si someisp.exemple offre le service "EM" avec le protocole "ProtB", il n'y a pas de raison de croire qu'il le fait au nom de exemple.com (car il n'y a pas un tel pointeur dans l'ensemble de RR NAPTR de exemple.com).

Il PEUT choisir quel protocole essayer d'abord sur la base de sa propre préférence, ou sur le rang de PREF dans le premier ensemble d'enregistrements NAPTR (c'est-à-dire, ceux pour le domaine cible désigné). Cependant, le protocole choisi DOIT être mentionné dans ce premier ensemble de RR NAPTR.

Il PEUT choisir de faire des résolutions DDDS simultanées pour plus d'un protocole, et dans ce cas, les exigences ci-dessus s'appliquent indépendamment pour chaque protocole. C'est-à-dire, il ne change pas de protocole à la mi-résolution.

3. Lignes directrices

3.1 Lignes directrices pour développeurs de protocole d'application

L'objet de S-NAPTR est de fournir aux développeurs de standard d'application un cadre plus puissant (que celui des RR SRV seuls) pour désigner les cibles de service, sans exiger que chaque norme de protocole d'application (ou service) définisse une application DDDS séparée.

Noter que cette approche est destinée spécifiquement à être utilisée lorsque il y a un sens à associer les services à des noms de domaine particuliers (par exemple, adresses de messagerie électronique, adresses SIP, etc.). Elle n'a pas pour but d'avoir toutes les sortes d'étiquettes transposées en un nom de domaine afin de l'utiliser.

Le présent document ne traite pas de la façon de choisir le domaine pour lequel la paire service+protocole est recherchée. D'autres conventions devront être définies sur la façon dont ceci peut être utilisé (par exemple, de nouvelles normes de messagerie peuvent définir quel domaine utiliser à partir de leurs URI ou comment revenir de foobar.exemple.com à exemple.com, si c'est applicable).

Bien que le présent document propose une application DDDS qui n'utilise pas toutes les caractéristiques des enregistrements de ressource NAPTR, il n'est pas destiné à impliquer que les résolveurs du DNS devraient échouer à mettre en œuvre tous les aspects du RR NAPTR standard. Une application DDDS est une convention à l'usage du client.
Le reste de cette section explique les éléments spécifiques que les développeurs de protocole doivent déterminer et documenter pour utiliser S-NAPTR.

3.1.1 Enregistrement des étiquettes de service et protocole d'application

Les développeurs de protocole d'application qui souhaitent faire usage de S-NAPTR doivent prendre des dispositions pour enregistrer toutes les étiquettes pertinentes de service d'application et de protocole d'application, comme décrit à la Section 7.

3.1.2 Définition des conditions de renouvellement d'essai/échec

Un autre important aspect qui doit être défini est le comportement attendu dans l'interaction avec les serveurs auxquels on accède via S-NAPTR. Précisément, dans quelles circonstances le client devrait-il réessayer une cible qui a été trouvée via S-NAPTR ? Que doit il considérer comme échec qui l'oblige à retourner au processus S-NAPTR pour déterminer la prochaine cible à desservir, qui par définition aura un rang de préférence inférieur.

Par exemple, si le client reçoit d'un serveur un message "connexion refusée", devrait-il réessayer pendant un certain temps (selon le protocole) ? Ou devrait-il essayer la cible suivante dans l'ordre de préférence dans la chaîne de résolution S-NAPTR ? Devrait-il seulement essayer la cible suivante dans l'ordre de préférence si il reçoit un message d'erreur permanent spécifique du protocole ?

Le plus important est de choisir un comportement attendu et de le documenter au titre de l'utilisation de S-NAPTR.

Comme noté précédemment, l'échec à fournir des accréditifs appropriés pour identifier le serveur comme étant d'autorité pour le domaine cible original est toujours considéré comme une condition d'échec.

3.1.3 Identification de serveur et prise de contact

Comme noté à la Section 8, l'utilisation du DNS pour la localisation de serveur augmente l'importance de l'utilisation de procédures d'accueil spécifiques du protocole pour déterminer et confirmer l'identité du serveur qui est finalement atteint.

Donc, les développeurs de protocole d'application qui utilisent S-NAPTR devraient identifier les mécanismes de l'accueil d'identification attendu lorsque le client se connecte à un serveur trouvé au moyen de S-NAPTR.

3.2 Lignes directrices pour les administrateurs de domaine

Bien que S-NAPTR vise à fournir une application "directe" de DDDS et utilise les enregistrements NAPTR, il est toujours possible de créer des chaînes et des dépendances très complexes avec les enregistrements NAPTR et SRV.

Donc, les administrateurs de domaines sont invités à utiliser S-NAPTR avec autant de réserve que possible tout en réalisant quand même leurs objectifs de conception de service.

Le jeu complet d'enregistrements de ressource NAPTR, SRV, et A "accessible" par le processus S-NAPTR pour un service d'application particulier peut être vu comme une "arborescence". Chaque RR NAPTR qui est restitué pointe sur plus d'enregistrements NAPTR ou SRV ; chaque enregistrement SRV pointe sur plusieurs recherches d'enregistrement A. Quand bien même un client particulier peut "élaguer" l'arborescence pour n'utiliser que les enregistrements qui se réfèrent aux protocoles d'application pris en charge par le client, l'arborescence peut être assez fournie, et reparcourir l'arborescence à la recherche des autres cibles peut devenir coûteux si l'arbre a de nombreuses branches.

Donc,

- moins de branches est meilleur : pour les enregistrements NAPTR comme SRV, fournir différentes cibles avec des préférences variées lorsque approprié (par exemple, pour fournir des services de sauvegarde) mais ne pas chercher des raisons pour en fournir plus ;
- moins profond est meilleur : éviter d'utiliser des enregistrements NAPTR pour "renommer" des services au sein d'une zone. Utiliser les enregistrements NAPTR pour identifier les services hébergés ailleurs (c'est-à-dire, lorsque on ne peut raisonnablement pas fournir d'enregistrements SRV dans sa propres zone).

3.3 Lignes directrices pour les rédacteurs de logiciel client

Pour bien comprendre DDDS/NAPTR, un développeur doit lire la [RFC3401]. Cependant, l'aspect le plus important à garder en mémoire est que si l'application ne peut pas réussir à se connecter à une cible, on attend de l'application qu'elle continue à travers l'arborescence S-NAPTR pour essayer des solutions de remplacement (moins préférées).

4. Illustrations

4.1 Cas d'utilisation

Les cas d'utilisation de base pour lesquels S-NAPTR a été développé sont les suivants :

- o Découverte de service au sein d'un domaine. Par exemple, ce peut être utilisé pour trouver le serveur"d'autorité" pour un certain type de service au sein d'un domaine (voir l'exemple spécifique au paragraphe 4.2).
- o Plusieurs protocoles. C'est déjà courant aujourd'hui car de nouveaux services d'application sont définis, et c'est un problème croissant. Il inclut le cas de messagerie extensible (un service hypothétique) qui peut être offerte avec plusieurs protocoles (voir au paragraphe 4.3).
- o Hébergement à distance. Chacun des cas d'utilisation ci-dessus s'applique au sein de l'administration d'un seul domaine. Cependant, un opérateur de domaine peut choisir d'engager une autre organisation pour fournir un service d'application. Voir au paragraphe 4.4 un exemple qui ne peut pas être servi par des enregistrements SRV seuls.

4.2 Découverte de service au sein d'un domaine

Il y a des occasions où il est utile d'être capable de déterminer le serveur"d'autorité" pour un certain service d'application au sein d'un domaine. C'est de la "découverte", car il n'y a pas de connaissance a priori de si ou où le service est offert ; il est donc important de déterminer la localisation et les caractéristiques du service offert.

Par exemple, il y a une discussion croissante sur un mécanisme générique pour localiser les clés ou les certificats associés à une application particulière (serveurs) qui fonctionnent dans (ou pour) un domaine particulier. Ce qui suit est un cas hypothétique pour mémoriser des données de clé ou certificat d'application pour un certain domaine : les prémisses sont qu'un service de registre d'accréditifs (CredReg) a été défini comme un service de nœud feuille qui détient les clés/certificats pour les serveurs gérés par (ou pour) le domaine. On suppose que plus d'un protocole est disponible pour fournir le service pour un domaine particulier. Cette approche fondée sur DDDS est utilisée pour trouver le serveur CredReg qui détient les informations.

Donc, l'ensemble d'enregistrements NAPTR pour chatpensant.exemple pourrait ressembler à :

```
chatpensant.exemple.
;;              ordr    préférenc   fanions
                e       e
IN NAPTR        100     10          ""      "CREDREG:ldap:iris.beep"    ( ; service
                                            ""                            ; regexp
                                            leserveur.chatpensant.exemple.  ; remplacement
```

Noter que le service d'application pourrait être offert dans un autre domaine en utilisant un ensemble différent de protocoles d'application :

```
autredomaine.exemple.
;;              ordr    préférenc   fanions
                e       e
IN NAPTR        100     10          ""      "CREDREG:iris.lwz:iris.beep"  ( ; service
                                            ""                              ; regexp
                                            foo.autredomaine.exemple.       ; remplacement
                                                                          )
```

4.3 Protocoles multiples

La messagerie extensible, un service d'application hypothétique, sera utilisée à des fins d'illustration. (Pour un exemple de réel service d'application avec plusieurs protocoles, voir les [RFC3982] et [RFC3983]). En supposant que "EM" a été enregistré comme service d'application, cette application DDDS pourrait être utilisée pour déterminer les services disponibles à livrer à une cible.

Deux caractéristiques particulières de cette messagerie extensible hypothétique devraient être notées :

1. Un service de passerelle est supposé ponter les communications à travers les protocoles.
2. Des serveurs de messagerie extensible vont probablement fonctionner pour sortir d'un domaine différent de celui de l'adresse de messagerie extensible, et des serveurs de protocoles différents peuvent être offerts par des organisations indépendantes.

Par exemple, "chatpensant.exemple" peut prendre en charge ses propres serveurs pour le protocole de messagerie extensible "ProtA" mais s'appuyer sur une source externe à partir de "exemple.com" pour les serveurs "ProtC" et "ProtB".

En utilisant cette approche fondée sur DDDS, chatpensant.exemple peut indiquer un rang de préférence pour les différents types de serveurs pour le service de messagerie extensible, et donc la source externe peut indépendamment ordonner les préférences et les serveurs. Cette indépendance n'est pas réalisable par l'utilisation des seuls enregistrements SRV.

Donc, pour trouver les services EM pour chatpensant.exemple, les enregistrements NAPTR pour chatpensant.exemple sont restitués :

```
chatpensant.exemple.
;;              ordre   préférence  fanion
                                    s
IN NAPTR        100     10          "s"     "EM:ProtA"                      (   ; service
                                            ""                                  ; regexp
                                            _ProtA._tcp.chatpensant.exemple.    ; remplacement
                                                                            )
IN NAPTR        100     20          "s"     "EM:ProtB"                      (   ; service
                                            ""                                  ; regexp
                                            _ProtB._tcp.exemple.com.            ; remplacement
                                                                            )
IN NAPTR        100     30          "s"     "EM:ProtC"                      (   ; service
                                            ""                                  ; regexp
                                            _ProtC._tcp.exemple.com             ; remplacement
                                                                            )
```

Ensuite, les administrateurs de exemple.com peuvent gérer les ordres de préférence des serveurs qu'ils utilisent pour prendre en charge le service ProtB:

```
_ProtB._tcp.exemple.com.
;;          Préf    Poids   Accès   Cible
IN SRV      10      0       10001   bigiron.exemple.com.
IN SRV      20      0       10001   backup.em.exemple.com.
IN SRV      30      0       10001   nuclearfallout.australia-isp.exemple.
```

4.4 Hébergement à distance

Dans l'exemple d'hébergement de message instantané du paragraphe 4.3, le propriétaire du service (chatpensant.exemple) avait à héberger des pointeurs sur les enregistrements SRV du service d'hébergement dans le domaine chatpensant.exemple.

Une meilleure approche est d'avoir un RR NAPTR dans le domaine chatpensant.exemple qui pointe sur tous les services hébergés. Le domaine d'hébergement a des enregistrements NAPTR pour chaque service pour les transposer en tout hôte local qu'il choisit (cela peut changer de temps en temps).

```
chatpensant.exemple.
;;              ordre   préférence  fanions
IN NAPTR        100     10          "s"     "EM:ProtA"                      (   ; service
                                            ""                                  ; regexp
                                            _ProtA._tcp.chatpensant.exemple.    ; remplacement
                                                                            )
IN NAPTR        100     20          ""      "EM:ProtB:ProtC"                (   ; service
                                            ""                                  ; regexp
                                            chatpensant.exemple.com.            ; remplacement
                                                                            )
```

Ensuite, les administrateurs de exemple.com peuvent ouvrir les protocoles d'application individuels et gérer l'ordre de préférence des serveurs qu'ils utilisent pour prendre en charge le service ProtB (comme précédemment) :

```
chatpensant.exemple.com.
;;              ordre     préférence     fanions
IN NAPTR    100       10             "s"        "EM:ProtC"                  ( ; service
                                                  ""                            ; regexp
                                                  _ProtC._tcp.exemple.com.      ; remplacement
                                                                              )
IN NAPTR    100       20             "s"        "EM:ProtB"                  ( ; service
                                                  ""                            ; regexp
                                                  _ProtB._tcp.exemple.com.      ; remplacement
                                                                              )

_ProtC._tcp.exemple.com.
 ;;             Préf      Poids  Accès    Cible
IN SRV      10        0      10001    bigiron.exemple.com.
IN SRV      20        0      10001    backup.em.exemple.com.
IN SRV      30        0      10001    nuclearfallout.australia-isp.exemple.
```

4.5 Ensembles de RR NAPTR

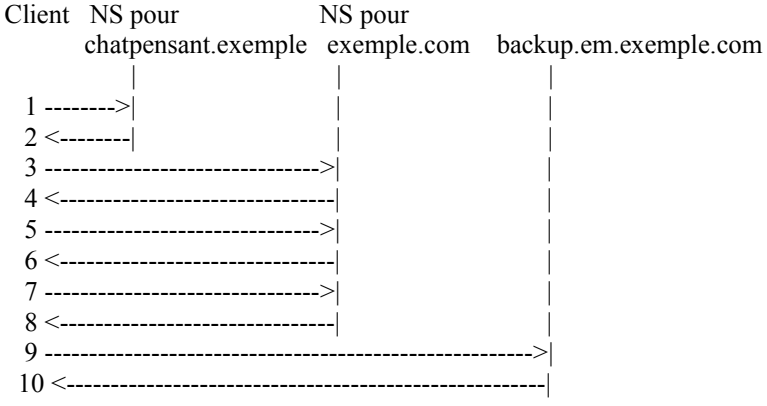
Noter que les paragraphes précédents supposent qu'il y avait un service disponible (via S-NAPTR) par domaine. Souvent, ce ne sera pas le cas. En supposant que chatpensant.exemple avait le service CredReg établi comme décrit au paragraphe 4.2 et avait le service de messagerie extensible établi comme décrit au paragraphe 4.4, alors un client qui interroge l'ensemble de RR NAPTR à partir de chatpensant.com obtiendrait la réponse suivante :

```
chatpensant.exemple.
;;              ordre     préférence     fanions
IN NAPTR    100       10             "s"        "EM:ProtA"                  (  ; service
                                                  ""                            ; regexp
                                                  _ProtA._tcp.chatpensant.exemple.  ; remplacement
                                                                              )
IN NAPTR    100       20             ""         "EM:ProtB:ProtC"            ( ; service
                                                  ""                            ; regexp
                                                  chatpensant.exemple.com.      ; remplacement
                                                                              )
IN NAPTR    200       10             ""         "CREDREG:ldap:iris-beep"    (  ; service
                                                  ""                            ; regexp
                                                  bouncer.chatpensant.exemple.  ; remplacement
                                                                              )
```

En les triant par "ORDER" croissant, le client va chercher à travers les chaînes SERVICE à déterminer si il y avait un RR NAPTR qui correspondait au service d'application qu'il recherche, avec un protocole d'application qu'il pourrait utiliser. Le client utiliserait le premier (la plus faible PREF) enregistrement qui correspond pour continuer.

4.6 Exemple de diagramme de séquence

Considérons l'exemple du paragraphe 4.3. Visuellement, la séquence d'étapes requises pour que le client atteigne le serveur final pour un service "ProtB" pour EM pour le domaine chatpensant.exemple est comme suit :



```
 11 ------------------------------------------------------>|
 12 <------------------------------------------------------|
(...)
```

1. Le serveur de noms (NS, *name server*) pour chatpensant.exemple est atteint par une demande pour tous les enregistrements NAPTR.
2. Le serveur répond par les enregistrements NAPTR montrés au paragraphe 4.3.
3. Le second enregistrement NAPTR satisfait aux critères désirés ; il a un fanion "s" et des champs de remplacement de "_ProtB._tcp.exemple.com". Donc le client cherche les enregistrements SRV pour cette cible, faisant en fin de compte la demande au NS pour exemple.com.
4. La réponse comporte les enregistrements SRV mentionnés au paragraphe 4.3.
5. Le client tente d'atteindre le serveur avec la plus faible PREF dans la liste de SRV – cherchant l'enregistrement A pour la cible de l'enregistrement SRV (bigiron.exemple.com).
6. Le NS exemple.com répond par un message d'erreur – pas de telle machine !
7. Le client tente d'atteindre le second serveur dans la liste de SRV et cherche l'enregistrement A pour backup.em.exemple.com.
8. Le client obtient l'enregistrement A avec l'adresse IP pour backup.em.exemple.com du NS de exemple.com.
9. Le client se connecte à cette adresse IP, sur l'accès 10001 (provenant de l'enregistrement SRV), en utilisant ProtB sur tcp.
10. Le serveur répond avec un message "OK".
11. Le client utilise ProtB pour demander si ce serveur a des accréditifs pour fournir le service pour le domaine d'origine (chatpensant.exemple)
12. Le serveur répond, et le reste est de la surveillance d'erreur.

5. Motivation et discussion

De plus en plus, les normes de protocole d'application utilisent les noms de domaines pour identifier les serveurs cibles et stipulent que les clients devraient chercher les enregistrements de ressource SRV pour déterminer l'hôte et l'accès qui fournissent le serveur. Cela permet une distinction entre désigner un service d'application cible et héberger réellement le serveur. Cela augmente aussi la souplesse en hébergeant le service cible, comme suit :

- Le serveur peut être géré par une organisation complètement différente sans avoir à faire la liste des détails de l'établissement DNS de cette organisation (les SRV).
- Plusieurs instances peuvent être établies (par exemple, pour l'équilibrage de charge ou les secondaires).
- Il peut être déplacé de temps en temps sans perturber l'accès des clients, etc.

L'approche est assez utile, mais le paragraphe 5.1 explique quelques une de ses limitations inhérentes.

C'est-à-dire, bien que les enregistrements SRV puissent être utilisés pour transposer d'un nom de service et protocole spécifiques pour un domaine spécifique en un serveur spécifique, les enregistrements SRV sont limités à une seule couche d'adressage et sont concentrés sur l'administration de serveur plutôt que sur la dénomination d'application. De plus, bien que la spécification de DDDS et l'utilisation de NAPTR permettent plusieurs niveaux de redirection avant que soit localisée la machine de serveur cible avec un enregistrement SRV, cette proposition n'exige qu'un sous ensemble de NAPTR strictement lié aux noms de domaines, sans faire usage du champ REGEXP de NAPTR. Ces restrictions rendent le processus de résolution du client plus prévisible et efficace qu'il ne le serait avec des utilisations potentielles d'enregistrements NAPTR. C'est ce qu'on appelle "S-NAPTR", une utilisation directe des enregistrements NAPTR.

5.1 Pourquoi pas juste des enregistrements SRV ?

Une question attendue à ce point est : ceci est si similaire par sa structure aux enregistrements SRV, pourquoi fait on cela avec DDDS/NAPTR ?

Les limitations du SRV incluent ce qui suit :

- SRV fournit une seule couche d'adressage ; le résultat d'une recherche de SRV est un nouveau nom de domaine pour lequel le RR A est à trouver.
- L'objet d'un SRV est de traiter des questions d'administration de serveur individuel, non de fournir une désignation d'application : comme il est dit dans la [RFC2782], "le RR SRV permet aux administrateurs d'utiliser plusieurs serveurs pour un seul domaine, de déplacer des services d'un hôte à un autre avec peu de confusion, et de désigner certains hôtes comme serveurs principaux pour un service et d'autres comme sauvegarde".
- Des serveurs cibles par "service" (par exemple, "ldap") et "protocole" (par exemple, "tcp") dans un certain domaine. La définition de ces termes implique des choses spécifiques (par exemple, que le protocole devrait être UDP ou TCP) sans

autre précision. Les restrictions sur UDP et TCP sont insuffisantes pour les utilisations qu'on décrit ici.

La réponse de base est que les enregistrements SRV fournissent des transpositions des noms de protocoles à des hôtes et des accès. Les cas d'utilisation décrits ici exigent une couche supplémentaire – provenant d'une étiquette de service à des serveurs qui peuvent être hébergés dans des domaines administratifs différents. On pourrait tordre le SRV pour dire que la prochaine recherche pourrait être quelque chose d'autre qu'un enregistrement d'adresse, mais ceci est plus complexe qu'il n'est nécessaire pour la plupart des applications de SRV.

5.2 Pourquoi pas juste des enregistrements NAPTR ?

C'est une question complexe. Les enregistrements NAPTR ne peuvent pas apparaître tout seuls ; voir la [RFC3401]. Ils doivent faire partie d'une application DDDS.

L'objet ici est de définir un seul mécanisme commun (l'application DDDS) pour utiliser le NAPTR alors que tout ce qu'on demande est une simple localisation de service fondée sur le DNS. Ce devrait être facile à utiliser pour les applications – quelques simples enregistrements de l'IANA, et c'est tout.

NAPTR a aussi des outils très puissants pour exprimer les règles "rewrite" *(réécriture)*. Cette puissance (==complexité) rend nerveux certains concepteurs de protocoles et administrateurs de services. Le souci est que ces réécritures peuvent se traduire en des ensembles de règles inintelligibles, comme des paquets de nouilles, qui sont difficiles à vérifier et à administrer.

L'application DDDS proposée utilise spécifiquement un sous ensemble des capacités de NAPTR. Seules les expressions "replacement" sont permises, pas les expressions "regular".

6. Définition formelle d'application <Application Service Location> de DDDS

Cette section définit formellement l'application DDDS, comme décrit dans la [RFC3401].

6.1 Chaîne unique pour l'application

La chaîne unique pour l'application est l'étiquette de domaine pour laquelle est recherché un serveur d'autorité pour un service particulier.

6.2 Première règle bien connue

La "première règle bien connue" est l'identité – c'est-à-dire, le résultat de la règle est la chaîne unique pour l'application, l'étiquette de domaine pour laquelle est recherché le serveur d'autorité pour un service particulier.

6.3 Résultats attendus

Le résultat attendu de cette application est l' information nécessaire pour qu'un client se connecte au ou aux serveurs d'autorité (hôte, accès, protocole) pour un service d'application particulier dans un certain domaine.

6.4 Fanions

Cette application DDDS utilise seulement deux des fanions définis pour l'application de résolution d'URI/URN [RFC3404] : "S" et "A". Aucun autre fanion n'est valide.

Tous deux sont pour les recherches terminales. Cela signifie que la règle est la dernière et que le fanion détermine ce que devrait être la prochaine étape. Le fanion "S" signifie que le résultat de cette règle est une étiquette de domaine pour laquelle existent un ou plusieurs enregistrements SRV [RFC2782]. "A" signifie que le résultat de la règle est un nom de domaine et devrait être utilisé pour rechercher des enregistrements d'adresse pour ce domaine.

En cohérence avec l'algorithme DDDS, si la chaîne Fanions est vide, la prochaine recherche est pour un autre enregistrement NAPTR (pour la cible de remplacement).

6.5 Paramètres de service

Les paramètres de service pour cette application prennent la forme d'une chaîne de caractères qui suit cet ABNF [RFC2234] :

```
service-parms = [ [app-service] *(":" app-protocol)]
app-service   = experimental-service  / iana-registered-service
app-protocol  = experimental-protocol / iana-registered-protocol
experimental-service      = "x-" 1*30ALPHANUMSYM
experimental-protocol     = "x-" 1*30ALPHANUMSYM
iana-registered-service   = ALPHA *31ALPHANUMSYM
iana-registered-protocol  = ALPHA *31ALPHANUM
ALPHA         =  %x41-5A / %x61-7A   ; A-Z / a-z
DIGIT         =  %x30-39 ; 0-9
SYM           =  %x2B / %x2D / %x2E  ; "+" / "-" / "."
ALPHANUMSYM   =  ALPHA / DIGIT / SYM
; les étiquettes app-service et app-protocol sont limitées à 32 caractères et doivent commencer par un caractère
; alphabétique. Les service-parms sont considérés comme insensibles à la casse. ;
```

Donc, les paramètres de service peuvent consister en une chaîne vide, un app-service, ou un app-service avec une ou plusieurs spécifications app-protocol séparées par le symbole ":".

Noter que ceci est similaire, mais pas identique, à la syntaxe utilisée dans l'application DDDS URI [RFC3404]. La base de données DNS de DDDS exige que chaque application DDDS définisse la syntaxe des chaînes de service admissibles. La syntaxe ici est étendue pour permettre les caractères qui sont valides dans tout nom de schéma d'URI (voir la [RFC2396]). Comme "+" (le séparateur utilisé dans la chaîne de paramètres de service de la RFC3404) est un caractère admis pour les noms de schéma d'URI, ":" est choisi ici comme séparateur.

6.5.1 Services d'application

Le "app-service" doit être un service enregistré par l'IANA ; voir à la Section 7 les instructions pour l'enregistrement de nouvelles étiquettes de service d'application.

6.5.2 Protocoles d'application

Les identifiants de protocole valides pour la production "app-protocol" sont des protocoles standard, enregistrés ; voir à la Section 7 les instructions pour l'enregistrement de nouvelles étiquettes de protocole d'application.

6.6 Règles valides

Seules les règles de substitution sont permises pour cette application. C'est-à-dire qu'aucune expressions régulière n'est permise.

6.7. Bases de données valides

À présent une seule base de données DDDS est spécifiée pour cette application. La [RFC3403] spécifie qu'une base de données DDDS qui utilise les enregistrement de ressource NAPTR du DNS contient les règles "rewrite". Les clés pour cette base de données sont codées comme des noms de domaines.

La première règle bien connue produit un nom de domaine, et c'est la clé utilisée pour la première recherche. Les enregistrements NAPTR pour ce domaine sont demandés.

Les serveurs DNS PEUVENT interpréter les valeurs de fanion et utiliser ces informations pour inclure les enregistrements NAPTR, SRV, ou A appropriés dans la portion Informations supplémentaires du paquet DNS. Les clients sont invités à vérifier qu'il y a des informations supplémentaires mais ne sont pas obligés de le faire. Voir la Section Traitement des informations supplémentaires de la [RFC3403] pour plus d'informations sur les enregistrements NAPTR et la section Informations supplémentaires d'un paquet de réponse du DNS.

7. Considérations relatives à l'IANA

Le présent document demande deux registres à l'IANA : un pour les étiquettes de service d'application, et un pour les étiquettes de protocole d'application.

7.1 Registre d'étiquettes de service d'application de l'IANA

L'IANA a établi et tiendra un registre des étiquettes de service d'application S-NAPTR, faisant la liste d'au moins les informations suivantes pour chacune de ces étiquettes :

- Étiquette de service d'application : chaîne conforme au service enregistré par l'IANA défini au paragraphe 6.5.

- Publication de définition : la RFC utilisée pour définir l'étiquette de service d'application, comme défini dans le processus d'enregistrement, ci-dessous.

Un enregistrement initial d'étiquettes de service d'application est contenu dans la [RFC3982].

7.2. Registre d'étiquettes de protocole d'application de l'IANA

L'IANA a établi et tiendra un registre des étiquettes de protocoles d'application pour S-NAPTR, faisant la liste d'au moins les informations suivantes pour chacune de ces étiquettes :

- Étiquette de protocole d'application : chaîne conforme au protocole enregistré par l'IANA définie au paragraphe 6.5.

- Publication de définition : la RFC utilisée pour définir l'étiquette de protocole d'application, comme définie dans le processus d'enregistrement, ci-dessous.

Un enregistrement initial d'étiquette de protocole d'application est défini dans la [RFC3983].

7.3 Processus d'enregistrement

Toutes les étiquettes de service et protocole d'application qui commencent par "x-" sont considérées comme expérimentales, et aucune disposition n'est prise pour empêcher la duplication de l'usage de la même chaîne. Les mises en œuvre les utilisent à leurs risques et périls.

Toutes les autres étiquettes de service et protocoles d'application sont enregistrées sur la base de l'option "spécification exigée" définie dans la [RFC2434], avec la stipulation supplémentaire que "spécification" est une RFC (de toute catégorie).

Aucune autre restriction ne pèse sur les étiquettes sauf qu'elles doivent se conformer à la syntaxe définie ci-dessous (paragraphe 6.5).

La RFC de définition doit clairement identifier et décrire, pour chaque étiquette enregistrée,
- l'étiquette de protocole ou service d'application,
- l'utilisation prévue,
- les considérations d'interopérabilité,
- les considérations de sécurité (voir la Section 8 de ce document pour la discussion des types de considérations applicables),
- toutes les publications en rapport pertinentes.

8. Considérations sur la sécurité

La sécurité de cette approche de la localisation de service d'application n'est pas meilleure que celle des interrogations du DNS le long du chemin. Si une d'elles est compromise, des enregistrements NAPTR et SRV bogués pourraient être insérés pour rediriger les clients sur des destinations imprévues. Ce problème n'est pas réservé à S-NAPTR (ou à NAPTR en général). On trouvera une discussion complète des menaces pour la sécurité qui relèvent du DNS dans la [RFC3833].

Pour se protéger contre les attaques visant le DNS, le DNS sécurisé (DNSSEC) [RFC4035] peut être utilisé pour s'assurer de la validité des enregistrements du DNS reçus.

Qu'on utilise ou non DNSSEC, les applications devraient définir une forme d'authentification de bout en bout pour s'assurer que la destination correcte a été atteinte. De nombreux protocoles d'application comme HTTPS, BEEP, et IMAP définissent les mécanismes de prise de contact nécessaires pour réaliser cette tâche. Les protocoles d'application nouveaux devraient prendre cela en compte et incorporer les mécanismes appropriés.

Le mécanisme de base fonctionne comme suit :

1. Durant une certaine portion de la prise de contact du protocole, le client envoie au serveur le nom original de la destination désirée (c'est-à-dire, aucune transformation qui pourrait avoir résulté des remplacements NAPTR, des SRV cibles, ou des changements de CNAME). Dans certains cas où le protocole d'application n'a pas une telle caractéristique mais où TLS peut être utilisé, il est possible d'utiliser l'extension TLS "server_name".

2. Le serveur renvoie au client un accréditif avec le nom approprié. Pour les certificats X.509, le nom serait soit dans le champ subjectDN, soit dans le champ subjectAltName. Pour Kerberos, le nom serait un nom de principe de service.

3. En utilisant la sémantique de confrontation définie par le protocole d'application, le client compare le nom dans l'accréditif avec le nom envoyé au serveur.

4. Si les noms correspondent, les accréditifs sont intègres, il y a une assurance raisonnable que le point d'extrémité correct a été atteint.

5. Le client et le serveur établissent un canal à l'intégrité protégée.

Noter que le présent document ne définit ni le mécanisme de prise de contact, ni les champs spécifiques de désignation d'accréditifs, ni la sémantique de confrontation des noms. Les définitions de S-NAPTR pour des protocoles d'application particuliers DOIVENT les définir.

9. Remerciements

Tous nos remerciements à Dave Blacka, Patrik Faltstrom, Sally Floyd, et Ted Hardie pour la discussion et les apports qui ont (heureusement !) provoqué la révision et des éclaircissements du présent document.

10. Références

10.1 Références normatives

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

[RFC2234] D. Crocker et P. Overell, "BNF augmenté pour les spécifications de syntaxe : ABNF", novembre 1997. (*Obsolète, voir* RFC5234)

[RFC2434] T. Narten et H. Alvestrand, "Lignes directrices pour la rédaction d'une section Considérations relatives à l'IANA dans les RFC", BCP 26, octobre, 1998. *(Rendue obsolète par la* RFC5226)

[RFC2782] A. Gulbrandsen, P. Vixie et L. Esibov, "Enregistrement de ressource DNS pour la spécification de la localisation des services (DNS SRV)", février 2000.

[RFC3401] M. Mealling, "Système de découverte dynamique de délégation (DDDS) Partie I : DDDS complet", octobre 2002. *(Info.)*

[RFC3403] M. Mealling, "Système de découverte dynamique de délégation (DDDS) Partie III : base de données du système de noms de domaines (DNS)", octobre 2002. *(P.S.)*

[RFC3404] M. Mealling, "Système de découverte dynamique de délégation (DDDS) Partie IV : Identifiants de ressource uniformes (URI)", octobre 2002. *(P.S.)*

10.2 Références pour information

[RFC2396] T. Berners-Lee, R. Fielding et L. Masinter, "Identifiants de ressource uniformes (URI) : Syntaxe générique",

août 1998. (*Obsolète, voir* RFC3986)

[RFC3833] D. Atkins, R. Austein, "Analyse des menaces contre le système des noms de domaines (DNS)", août 2004. *(Information)*

[RFC3982] A. Newton, M. Sanz, "IRIS : un type de registre de domaines (dreg) pour le service d'information des registres Internet (IRIS)", janvier 2005. *(P.S.)*

[RFC3983] A. Newton, M. Sanz, "Utilisation du service d'information des registres Internet (IRIS) sur le protocole extensible d'échange de blocs (BEEP)", janvier 2005. *(P.S.)*

[RFC4035] R. Arends et autres, "Modifications du protocole pour les extensions de sécurité du DNS", mars 2005.

Appendice A. Pseudo-pseudocode pour S-NAPTR

A.1 Trouver la première cible

En supposant que le client prend en charge un protocole pour un service d'application particulier, le pseudocode suivant décrit le processus prévu pour trouver la première cible (la meilleure) pour le client, en utilisant S-NAPTR.

```
cible = [domaine initial]
  naptr-done = faux

tandis que (non naptr-done)
   {
    NAPTR-RRset = [recherche DNS de RR NAPTR pour la cible]
    [trier NAPTR-RRset par ORDER, et PREF au sein de chaque ORDER]
    rr-done = faux
    cur-rr = [premier RR NAPTR]

tandis que (non rr-done)
      si ([champ SERVICE de cur-rr contient le service d'application et le protocole d'application])
        rr-done = vrai
        cible= [cible de REMPLACEMENT de RR NAPTR]
      autrement
        cur-rr = [prochain rr dans la liste]
      si (non vide [FLAG dans cur-rr])
        naptr-done = vrai
   }

  port = -1

  si ([FLAG dans cur-rr est "S"])
   {
    SRV-RRset = [recherche DNS des RR SRV pour la cible]
    [trier SRV-RRset sur la base de PREF]
    cible = [cible du premier RR de SRV-RRset]
    accès = [accès dans le premier RR de SRV-RRset]
   }

; maintenant, que ce soit un "S" ou un "A" dans le NAPTR, on a la cible pour une recherche d'enregistrement A

  hôte = [recherche DNS de la cible]

  retour (hôte, accès)
```

A.2 Trouver les cibles suivantes

Le pseudocode de l'Appendice A est conçu pour trouver la première (préférée) paire hôte-accès pour un service et protocole d'application particulier. Si, pour n'importe quelle raison, cette paire hôte-accès ne fonctionne pas (connexion refusée, erreur de niveau application) le client est supposé essayer le prochain hôte-accès dans l'arborescence S-NAPTR.

Le pseudocode ci-dessus ne permet pas de réessai ; une fois achevé, il perd tout le contexte de l'arborescence de S-NAPTR où il a terminé. Donc les rédacteurs de logiciel client pourraient :

- o mêler le protocole spécifique d'application avec la recherche DNS et le traitement de RRset décrit dans le pseudocode et continuer le traitement S-NAPTR si le code d'application ne réussit pas à connecter à une paire localisée d'hôte-accès ;
- o utiliser des rappels pour le traitement de S-NAPTR ; ou
- o utiliser un sous programme de résolution S-NAPTR qui trouve *tous* les serveurs valides pour le service et protocole d'application demandé pour le domaine d'origine et qui les fournit triés dans l'ordre d'essai de l'application.

Appendice B. Disponibilité de l'échantillon de code

L'échantillon de code Python pour la résolution de S-NAPTR est disponible à http://www.verisignlabs.com/pysnaptr-0.1.tgz

Adresse des auteurs

Leslie Daigle
VeriSign, Inc.
21355 Ridgetop Circle
Dulles, VA 20166
US
mél : leslie@verisignlabs.com ; leslie@chatpensant.com

Andrew Newton
VeriSign, Inc.
21355 Ridgetop Circle
Dulles, VA 20166
US
mél : anewton@verisignlabs.com

Déclaration complète de droits de reproduction

Remerciement
Le financement de la fonction d'édition des RFC est actuellement fourni par l'Internet Society