Groupe de travail Réseau
Request for Comments : 2845
RFC mise à jour : 1035
Catégorie : En cours de normalisation
Traduction Claude Brière de L'Isle

P. Vixie, ISC
O. Gudmundsson, NAI Labs
D. Eastlake 3rd, Motorola
B. Wellington, Nominum
mai 2000

Authentification de transaction de clé secrète pour DNS (TSIG))

Résumé
Le présent protocole permet une authentification au niveau de la transaction en utilisant des secrets partagés et un hachage unidirectionnel. Il peut être utilisé pour authentifier des mises à jour dynamiques comme provenant d'un client approuvé, ou pour authentifier des réponses comme provenant d'un serveur de noms approuvé récurrent.

Aucune disposition n'a été prise ici pour répartir les secrets partagés ; on pense qu'un administrateur de réseau configurera les serveurs de noms et les clients de façon statique en utilisant un mécanisme hors bande tel qu'un réseau furtif jusqu'à ce qu'un mécanisme automatique sûr de distribution de clés soit disponible.

1. Introduction

1.1 Le système des noms de domaines (DNS, *Domain Name System*) [RFC1034], [RFC1035] est un système de bases de données réparties hiérarchisées et dupliquées qui fournit des informations fondamentales pour le fonctionnement de l'Internet telles que la traduction de nom <=> adresse et des informations de traitement de la messagerie. Le DNS a récemment été étendu [RFC2535] pour fournir l'authentification de l'origine des données, et la distribution de clés publiques, sur la base de la cryptographie à clé publique et les signatures numériques fondées sur la clé publique. Pour rester pratique, cette forme de sécurité exige généralement une mise en antémémoire locale extensive des clés et le traçage de l'authentification à travers de multiples clés et signatures à une clé de pré-confiance configurée localement.

1.2 Une difficulté avec le schéma de la [RFC2535] est que les mises en œuvre courantes du DNS comportent de simples résolveurs de "bout" qui n'ont pas d'antémémoire. De tels résolveurs s'appuient normalement sur un serveur DNS à antémémoire sur un autre hôte. Il est impraticable pour ces résolveurs de bout d'effectuer l'authentification générale de la [RFC2535] et ils vont naturellement dépendre de leur serveur DNS à antémémoire pour effectuer de tels services pour eux. Faire cela en toute sécurité exige une communication sécurisée des interrogations et des réponses. La [RFC2535] fournit des signatures de transaction de clé publique pour prendre cela en charge, mais la génération de telles signatures est très coûteuse en calcul. En général, cela exige la même logique de clé publique complexe qui est impraticable pour les bouts. Le présent document spécifie l'utilisation d'un code d'authentification de message (MAC, *message authentication code*) précisément HMAC-MD5 (une fonction de hachage à clé) pour fournir un moyen efficace d'authentification point à point et de vérification d'intégrité pour les transactions.

1.3 Un second domaine où l'utilisation de mécanismes directs fondés sur la clé publique [RFC2535] peuvent être impraticables est l'authentification de demandes de mise à jour dynamiques [RFC2136]. La [RFC2535] traite des signatures de demandes mais avec la [RFC2535] elles exigent , comme les signatures de transaction, un chiffrement de clé publique coûteux en calcul et une logique d'authentification complexe. La mise à jour dynamique sécurisée du système des noms de domaine de la [RFC2137]) décrit comment différentes clés sont utilisées dans les zones à mise à jour dynamique. Les MAC fondés sur une clé secrète du présent document peuvent être utilisés pour authentifier les demandes de mise à jour du DNS aussi bien que les réponses de transaction, fournissant une solution de remplacement légère au protocole décrit par la [RFC2137].

1.4 Une autre utilisation de ce mécanisme est de protéger les transferts de zone. Dans ce cas, les données couvertes

seront la totalité du transfert de zone incluant tous les enregistrements glu envoyés. Le protocole décrit par la [RFC2535] ne protège pas les enregistrements glu et les enregistrements non signés sauf si SIG(0) (signature de transaction) est utilisé.

1.5 Le mécanisme d'authentification proposé dans le présent document utilise les clés secrètes partagées pour établir une relation de confiance entre deux entités. De telles clés doivent être protégées d'une façon similaire à celles des clés privées, de peur qu'un tiers usurpe l'identité d'une des parties prévues (avec des MAC falsifiés). Il y a un besoin urgent de fournir une authentification simple et efficace entre les clients et les serveurs locaux et cette proposition s'adresse à ce besoin. Cette proposition ne convient pas pour l'authentification générale de serveur à serveur pour les serveurs qui correspondent avec de nombreux autres serveurs, car la gestion de clés deviendrait peu maniable avec la croissance exponentielle du nombre de clés partagées. Mais elle convient pour de nombreux résolveurs sur des hôtes qui ne parlent qu'avec peu de serveurs récurrents.

1.6 Un serveur agissant comme un résolveur indirect à antémémoire -- un "transmetteur" dans l'usage courant -- pourrait utiliser l'authentification fondée sur la transaction lors de ses communications avec son petit nombre de serveurs "amont" préconfigurés. D'autres usages de l'authentification DNS par clé secrète et de possibles systèmes pour une distribution automatique de clé secrète pourront être proposés dans de futurs documents distincts.

1.7 Nouveaux numéros alloués

RRTYPE = TSIG (250)
ERROR = 0..15 (RCODE DNS)
ERROR = 16 (BADSIG)
ERROR = 17 (BADKEY)
ERROR = 18 (BADTIME)

1.8 Les mots clés "DOIT", "EXIGÉ", "DEVRAIT", "RECOMMANDE", et "PEUT" dans le présent document sont à interpréter comme décrit dans la [RFC2119].

2. Format de RR TSIG

2.1 Type de RR TSIG

Pour fournir l'authentification de clé secrète, on utilise un nouveau type de RR dont le mnémonique est TSIG et dont le code de type est 250. TSIG est un méta-RR et NE DOIT PAS être mis en antémémoire. Les RR TSIG sont utilisés pour l'authentification entre les entités du DNS qui ont établi une clé secrète partagée. Les RR TSIG sont calculés de façon dynamique pour couvrir une transaction DNS particulière et ne sont pas des RR DNS au sens usuel.

2.2 Calcul de TSIG

Comme les RR TSIG se rapportent à une demande/réponse DNS, il n'y a pas d'intérêt à les mémoriser ou les retransmettre, et donc le RR TSIG est éliminé une fois qu'il a été utilisé pour authentifier un message DNS. Le seul algorithme de résumé de message spécifié dans le présent document est "HMAC-MD5" (voir les [RFC1321], [RFC2104]). La mise en œuvre de l'algorithme "HMAC-MD5" est obligatoire pour l'interopérabilité. D'autres algorithmes pourront être spécifiés à une date ultérieure. Les noms et définitions des nouveaux algorithmes MOIVENT être enregistrés auprès de l'IANA. Tous les entiers multi-octet dans l'enregistrement TSIG sont envoyés dans l'ordre des octets du réseau (voir le paragraphe 2.3.2 de la [RFC1035]).

2.3 Format d'enregistrement

NOM C'est le nom de la clé utilisée dans la syntaxe des noms de domaines. Le nom devrait refléter les noms des hôtes et identifier de façon univoque la clé parmi un ensemble de clés que ces deux hôtes peuvent partager à tout moment. Si les hôtes A.site.example et B.example.net partagent une clé, les possibilités pour le nom de la clé incluent <id>.A.site.example, <id>.B.example.net, et <id>.A.site.example.B.example.net. Il devrait être possible que plus d'une clé soit en usage simultanément parmi un ensemble d'hôtes en interaction. Le nom a seulement besoin d'être significatif pour les hôtes en communication mais un nom mnémonique significatif comme ci-dessus est fortement recommandé.

Le nom peut être utilisé comme un indice local pour la clé impliquée et il est recommandé qu'il soit unique au monde. Lorsque une clé est juste partagée entre deux hôtes, son nom n'a en fait besoin d'être significatif que pour eux mais il est recommandé que le nom de la clé soit mnémonique et incorpore les noms du résolveur et du serveur dans cet ordre.

TYPE TSIG (250 : Transaction SIGnature)

CLASSE ANY

TTL 0

RdLen (variable)

RDATA

Nom de champ	**Type de données**	**Notes**
Nom d'algorithme	nom de domaine	nom de l'algorithme dans la syntaxe de nom de domaines.
Heure signée	u_int48_t	secondes depuis le 1 janvier 1970 00 h UTC.
Tolérance	u_int16_t	secondes d'erreur permises dans Heure signée.
Taille de MAC	u_int16_t	nombre d'octets dans le MAC.
MAC	flux d'octets	défini par Nom d'algorithme.
ID d'origine	u_int16_t	ID du message d'origine
Erreur	u_int16_t	RCODE étendu couvrant le traitement de la TSIG.
Autre Longueur	u_int16_t	longueur, en octets, de Autres données.
Autres données	flux d'octets	vide, sauf Erreur == BADTIME

2.4 Exemple

NOM HOST.EXAMPLE.
TYPE TSIG
CLASSE ANY
TTL 0
RdLen comme approprié
RDATA

Nom du champ	**Contenu**
Nom d'algorithme	SAMPLE-ALG.EXAMPLE.
Heure signée	853804800
Tolérance	300
Taille de MAC	comme approprié
MAC	comme approprié
ID d'origine	comme approprié
Erreur	0 (NOERROR)
Autre Longueur	0
Autres données	vide

3. Fonctionnement du protocole

3.1 Effets de l'ajout de TSIG aux messages sortants

Une fois que le message sortant a été construit, l'opération de résumé chiffré de message peut alors être effectuée. Le résumé de message résultant sera alors mémorisé dans une TSIG qui sera ajoutée à la section des données supplémentaires (le ARCOUNT est incrémenté pour refléter cela). Si l'enregistrement TSIG ne peut pas être ajouté sans causer la troncature du message, le serveur DOIT altérer la réponse de telle sorte qu'une TSIG puisse être incluse. Cette réponse ne comporte que la question et un enregistrement TSIG, et a le bit TC établi et le RCODE 0 (NOERROR). Le client DEVRAIT à ce moment réessayer la demande en utilisant TCP (selon le paragraphe 4.2.2 de la [RFC1035]).

3.2 Traitement de TSIG sur les messages entrants

Si un message entrant contient un enregistrement TSIG, il DOIT être le dernier enregistrement dans la section Informations supplémentaires. Il n'est pas permis qu'il y ait plusieurs enregistrements TSIG. Si un enregistrement TSIG est présent dans n'importe quelle autre position, le paquet est abandonné et une réponse avec le RCODE 1 (FORMERR) DOIT être retournée. À réception d'un message avec un RR TSIG correctement placé, le RR TSIG est copié dans une localisation sûre, retiré du message DNS, et décrémenté du ARCOUNT de l'en-tête du message DNS. À ce point est effectuée l'opération de résumé chiffré du message. Si le nom de l'algorithme ou le nom de la clé est inconnu du receveur, ou si les résumés de message ne correspondent pas, la totalité du message DNS DOIT être éliminée. Si le message est une interrogation, une réponse avec le RCODE 9 (NOTAUTH) DOIT être renvoyée à l'origine avec TSIG ERROR 17 (BADKEY) ou TSIG ERROR 16 (BADSIG). Si aucune clé n'est disponible pour signer ce message, il DOIT être envoyé non signé (Taille de MAC == 0 et MAC vide). Un message au journal d'opérations du système DEVRAIT être généré, pour avertir le personnel de surveillance d'un possible incident de sécurité en développement. Il faut veiller à s'assurer que des inscriptions de ce type n'ouvrent pas la porte à une attaque de déni de service contre le système.

3.3 Valeurs horaires utilisées dans les calculs de TSIG

Les données résumées comportent les deux valeurs de temporisateurs de l'en-tête de TSIG afin de se défendre contre les attaques en répétition. Si cela n'était pas fait, un attaquant pourrait répéter de vieux messages mais mettre à jour les champs "Heure signée" et "Tolérance" pour faire croire que le message est nouveau. Ces données sont nommées "Temporisateurs TSIG", et pour les besoins du calcul du résumé, elles sont invoquées sous leur format "réseau", dans l'ordre suivant : d'abord Heure signée, puis Tolérance. Par exemple :

Nom de champ	Valeur	Format réseau	Signification
Heure signée	853804800	00 00 32 e4 07 00	mardi 21 janvier 1997 à 00:00:00
Tolérance	300	01 2C	5 minutes

3.4 Variables TSIG et portée

Lors de la génération ou de la vérification du contenu d'un enregistrement TSIG, les données suivantes sont résumées, dans l'ordre des octets du réseau, comme approprié :

3.4.1 Message DNS

C'est la totalité d'un message DNS complet en format réseau, avant l'ajout du RR TSIG à la section de données supplémentaires et avant que le champ ARCOUNT de l'en-tête du message DNS ait été incrémenté pour contenir le RR TSIG. Si l'identifiant du message diffère de celui du message original, l'identifiant du message original est substitué à l'identifiant de message. Cela pourrait arriver lors de la transmission d'une demande de mise à jour dynamique, par exemple.

3.4.2 Variables TSIG

Source	Nom du champ	Notes
TSIG RR	NOM	Nom de clé, en forme canonique du réseau
TSIG RR	CLASSE	(Toujours ANY dans la spécification actuelle)
TSIG RR	TTL	(Toujours 0 e DNS Message Header's)
TSIG RDATA	Nom d'algorithme	en format canonique du réseau
TSIG RDATA	Heure signée	dans l'ordre des octets du réseau
TSIG RDATA	Tolérance	dans l'ordre des octets du réseau
TSIG RDATA	Erreur	dans l'ordre des octets du réseau
TSIG RDATA	Autre Longueur	dans l'ordre des octets du réseau
TSIG RDATA	Autres données	exactement comme transmis

La RDLEN du RR et la Longueur du MAC de RDATA ne sont pas incluses dans le hachage car il n'est pas garanti qu'elles puissent être connues avant la génération du MAC.

Le champ Identifiant d'origine n'est pas inclus dans cette section, car il a déjà été substitué à l'ID de message dans l'en-tête DNS et haché.

Pour chaque type d'étiquette, il doit y avoir un "format canonique du réseau" défini qui spécifie comment exprimer sans ambiguïté une étiquette. Pour le type d'étiquette 00, c'est défini dans la [RFC2535], pour le type d'étiquette 01, c'est défini dans la [RFC2673]. L'utilisation de types d'étiquettes autres que 00 et 01 n'est pas défini pour la présente spécification.

3.4.3 MAC de demande

Lors de la génération du MAC à inclure dans une réponse, le MAC de demande doit être inclus dans le résumé. Le MAC de la demande est résumé en format réseau, incluant les champs suivants :

Champ	Type	Description
Longueur de MAC	u_int16_t	dans l'ordre des octets du réseau
Données du MAC	flux d'octets	exactement comme transmis

3.5 Bourrage

Les composants résumés sont fournis à la fonction de hachage comme un flux d'octets continu sans bourrage entre champs.

4. Détails du protocole

4.1 Génération de TSIG sur les demandes

Le client effectue l'opération de résumé de message et ajoute un enregistrement TSIG à la section de données supplémentaires et transmet la demande au serveur. Le client DOIT mémoriser le résumé de message réalisé à partir de la demande tout en attendant la réponse. Les composants du résumé pour une demande sont :

Message DNS (demande)
Variables TSIG (demande)

Noter que certains serveurs de noms plus anciens ne vont pas accepter les demandes avec une section de données supplémentaires non vide. Les clients DEVRAIENT seulement tenter des transactions signées avec des serveurs qui sont connus pour prendre en charge TSIG et partagent des clés secrètes avec le client – ainsi, il n'y a pas de problème en pratique.

4.2 TSIG sur les réponses

Lorsque un serveur a généré une réponse à une demande signée, il signe la réponse en utilisant le même algorithme et clé. Le serveur NE DOIT PAS générer une réponse signée à une demande non signée. Les composants du résumé sont :

MAC de demande
Message DNS (réponse)
Variables TSIG (réponse)

4.3 TSIG sur retour d'erreur TSIG

Lorsque un serveur détecte une erreur se rapportant à une clé ou à un MAC, le serveur DEVRAIT renvoyer un message d'erreur non signé (taille de MAC == 0 et MAC vide). Si une erreur est détectée qui se rapporte à la période de validité de la TSIG, le serveur DEVRAIT renvoyer un message d'erreur signé. Les composants du résumé sont :

MAC de demande (si le MAC de demande est validé)
Message DNS (réponse)
Variables TSIG (réponse)

La raison pour laquelle dans certains cas la demande n'est pas incluse dans ce résumé est de rendre possible au client de vérifier l'erreur. Si l'erreur n'est pas une erreur de TSIG, la réponse DOIT être générée comme spécifié en [4.2].

4.4 TSIG sur connexion TCP

Une session DNS TCP peut inclure plusieurs enveloppes DNS. Ceci est, par exemple, d'usage courant pour les transferts de zone. Utiliser TSIG sur une telle connexion peut protéger la connexion d'une agression et assure la protection d'intégrité des

données. La TSIG DOIT être incluse dans la première et la dernière enveloppe DNS. Elle peut facultativement être placée sur toute enveloppe intermédiaire. Il est coûteux de l'inclure sur chaque enveloppe, mais elle DOIT être placée sur au moins chaque 100e enveloppe. La première enveloppe est traitée comme une réponse standard, et les messages suivants ont les composants de résumé suivants :

Avant résumé (en cours)
Messages DNS (tout message non signé depuis la dernière TSIG)
Temporisateurs TSIG (message en cours)

Cela permet au client de détecter rapidement quand la session a été altérée; à quel point il peut clore la connexion et réessayer. Si une vérification de TSIG du client échoue, le client DOIT clore la connexion. Si le client ne reçoit pas assez fréquemment les enregistrements TSIG (comme spécifié ci-dessus) il DEVRAIT supposer que la connexion a été capturée et il DEVRAIT clore la connexion. Le client DEVRAIT traiter cela de la même façon qu'il le ferait de tout autre transfert interrompu (bien que le comportement exact ne soit pas spécifié).

4.5 Vérifications de TSIG par le serveur

À réception d'un message, le serveur va vérifier si il y a un RR TSIG. Si il en existe un, le serveur DOIT retourner un RR TSIG dans la réponse. Le serveur DOIT effectuer les vérifications suivantes dans l'ordre : vérifier KEY, vérifier les valeurs de TIME, vérifier le MAC.

4.5.1 Vérification de KEY et traitement des erreurs

Si un serveur non transmetteur ne reconnaît pas la clé utilisée par le client, le serveur DOIT générer une réponse d'erreur avec le RCODE 9 (NOTAUTH) et la TSIG ERROR 17 (BADKEY). Cette réponse DOIT être non signée comme spécifié en [4.3]. Le serveur DEVRAIT inscrire l'erreur dans le journal des événements.

4.5.2 Vérification de TIME et traitement des erreurs

Si l'heure du serveur est en dehors de l'intervalle horaire spécifié par la demande (qui est : Heure signée, plus/moins Tolérance) le serveur DOIT générer une réponse d'erreur avec le RCODE 9 (NOTAUTH) et une TSIG ERROR 18 (BADTIME). Le serveur DEVRAIT aussi mettre en antémémoire la plus récente valeur d'heure signée dans un message généré par une clé, et DEVRAIT retourner BADTIME si un message reçu ultérieurement a une valeur d'heure signée antérieure. Une réponse indiquant une erreur BADTIME DOIT être signée par la même clé que la demande. Elle DOIT inclure l'heure actuelle du client dans le champ Heure signée, l'heure actuelle du serveur (un u_int48_t) dans le champ Autres données, et 6 dans le champ Longueur des autres données. Cela est fait de telle sorte que le client puisse vérifier un message avec une erreur BADTIME sans qu'échoue la vérification du fait d'une autre erreur BADTIME. Les données signées sont spécifiées en [4.3]. Le serveur DEVRAIT inscrire l'erreur dans le journal des événements.

4.5.3 Vérification de MAC et traitement des erreurs

Si il y a échec de la vérification de TSIG, le serveur DOIT générer une réponse d'erreur comme spécifié en [4.3] avec le RCODE 9 (NOTAUTH) et une TSIG ERROR 16 (BADSIG). Cette réponse DOIT être non signée comme spécifié en [4.3]. Le serveur DEVRAIT inscrire l'erreur dans le journal des événements.

4.6 Traitement de la réponse par le client

Lorsque un client reçoit une réponse d'un serveur et s'attend à voir une TSIG, il vérifie d'abord si le RR TSIG est présent dans la réponse. Autrement, la réponse est traitée comme ayant une erreur de format et est éliminé. Le client extrait alors la TSIG, ajuste le ARCOUNT, et calcule le résumé chiffré de la même façon que le serveur. Si la TSIG n'est pas validée, cette réponse DOIT être éliminée, sauf si le RCODE est 9 (NOTAUTH) auquel cas le client DEVRAIT tenter de vérifier la réponse comme si elle était une réponse Erreur TSIG, comme spécifié en [4.3]. Un message qui contient un enregistrement TSIG non signé ou un enregistrement TSIG qui échoue à la vérification NE DEVRAIT PAS être considéré comme une réponse acceptable ; le client DEVRAIT inscrire une erreur dans le journal des événements et continuer d'attendre une réponse signée jusqu'à la péremption de la demande.

4.6.1 Traitement d'une erreur de clé

Si un RCODE sur une réponse est 9 (NOTAUTH), et si la réponse TSIG est validée, et si la clé de TSIG est différente de celle utilisée dans la demande, c'est alors une erreur de clé. Le client PEUT réessayer la demande en utilisant la clé spécifiée par le serveur. Cela ne devrait jamais arriver car un serveur NE DOIT PAS signer une réponse avec une clé différente de celle qui signait la demande.

4.6.2 Traitement des erreurs d'heure

Si le RCODE de réponse est 9 (NOTAUTH) et si la TSIG ERROR est 18 (BADTIME), ou si l'heure actuelle ne tombe pas dans la gamme spécifiée dans l'enregistrement TSIG, c'est alors une erreur d'heure. C'est une indication que les horloges du client et du serveur ne sont pas synchronisées. Dans ce cas, le client DEVRAIT enregistrer l'événement dans le journal. Les résolveurs du DNS NE DOIVENT PAS ajuster d'horloges chez le client sur la base des erreurs BADTIME, mais l'heure du serveur dans le champ Autres données DEVRAIT être enregistrée dans le journal.

4.6.3 Traitement des erreurs de MAC

Si le RCODE de réponse est 9 (NOTAUTH) et si la TSIG ERROR est 16 (BADSIG), c'est une erreur de MAC, et le client PEUT réessayer la demande avec un nouvel identifiant de demande mais il serait préférable d'essayer une clé partagée différente si il en est une disponible. Le client DEVRAIT garder trace du nombre d'erreurs de MAC associé à chaque clé. Les clients DEVRAIENT enregistrer cet événement.

4.7 Considérations particulières pour les serveurs transmetteurs

Un serveur qui agit comme serveur transmetteur d'un message DNS DEVRAIT vérifier l'existence d'un enregistrement TSIG. Si le nom sur la TSIG n'est pas celui d'un secret que le serveur partage avec l'origine du message, le serveur DOIT transmettre le message inchangé, y compris la TSIG. Si le nom de la TSIG est celui d'une clé que ce serveur partage avec l'origine du message, il DOIT traiter la TSIG. Si la TSIG réussit toutes les vérifications, le serveur transmetteur DOIT, si possible, inclure une TSIG de son cru, vers la destination ou le prochain transmetteur. Si aucune sécurité de transaction n'est disponible vers la destination et si la réponse a le fanion AD établi (voir la [RFC2535]) le transmetteur DOIT ôter le fanion AD avant d'ajouter la TSIG à la réponse.

5. Secrets partagés

5.1 Les clés secrètes sont des informations très sensibles et toutes les mesures disponibles devraient être prises pour les protéger sur tous les hôtes sur lesquels elles sont mémorisées. Généralement de tels hôtes doivent être protégés physiquement. Si ce sont des machines multi utilisateur, un grand soin devrait être apporté a empêcher l'accès d'utilisateurs non privilégiés au matériel de clés. Les résolveurs fonctionnent souvent sans privilège, ce qui signifie que tous les utilisateurs d'un hôte vont être capables de voir toutes les données de configuration utilisées par le résolveur.

5.2 Un serveur de noms fonctionne usuellement avec des privilèges, ce qui signifie que ses données de configuration n'ont pas besoin d'être visibles de tous les utilisateurs de l'hôte. Pour cette raison, un hôte qui met en œuvre l'authentification fondée sur la transaction devrait probablement être configuré avec un "résolveur de bout" et une antémémoire locale et un serveur de noms transmetteur. Cela pose un problème particulier pour la [RFC2136] qui s'appuie autrement sur les clients pour ne communiquer qu'avec les serveurs de noms d'autorité d'une zone.

5.3 L'utilisation de secrets partagés de fort aléa est essentielle pour la sécurité des TSIG. Voir dans la [RFC1750] une discussion de ce sujet. Le secret devrait au moins être aussi long que le résumé du message chiffré, c'est à dire, 16 octets pour HMAC-MD5 ou 20 octets pour HMAC-SHA1.

6. Considérations pour la sécurité

6.1 L'approche spécifiée ici est beaucoup moins coûteuse en calcul que les signatures spécifiées dans la [RFC2535]. Tant que la clé secrète partagée n'est pas compromise, une forte authentification est fournie pour le dernier bond, du serveur de nom local au résolveur de l'usager.

6.2 Les clés secrètes devraient être changées périodiquement. Si l'hôte client a été compromis, le serveur devrait suspendre l'utilisation de tous les secrets connus de ce client. Si possible, les secrets devraient être mémorisés sous forme chiffrée. Les secrets ne devraient jamais être transmis en clair sur un réseau. Le présent document ne traite pas la question de la distribution des secrets. Les secrets ne devraient jamais être partagés par plus de deux entités.

6.3 Ce mécanisme n'authentifie pas les données de la source, mais seulement leur transmission entre deux parties qui partagent un secret. Les données originales de la source peuvent provenir d'un maître de zone compromis ou peuvent être corrompues durant le transit d'un maître de zone authentique à un "transmetteur à antémémoire". Cependant, si le serveur effectue de bonne foi toutes les vérifications de sécurité de la [RFC2535] seules les données dont la sécurité a été vérifiée seront disponibles pour le client.

6.4 Une tolérance de valeur qui est trop importante peut laisser le serveur ouvert à des attaques de répétition. Une valeur de tolérance qui serait trop faible peut causer des défaillances si les machines ne sont pas synchronisées ou si il y a des délais inattendus dans le réseau. La valeur recommandée dans la plupart des situation est de 300 secondes.

7. Considérations relatives à l'IANA

On attend de l'IANA qu'elle crée et entretienne un registre des noms d'algorithmes à utiliser comme "Nom d'algorithme" comme défini au paragraphe 2.3. La valeur initiale devrait être "HMAC-MD5.SIG-ALG.REG.INT". Les noms d'algorithmes sont des chaînes de texte codé en utilisant la syntaxe d'un nom de domaine. Aucune structure autre que l'unicité des noms des différents algorithmes n'est requise lors de leur comparaison comme noms DNS, c'est-à-dire que la comparaison est insensible à la casse. Noter que la valeur initiale mentionnée ci-dessus n'est pas un nom de domaine, et donc n'a pas besoin d'être un nom enregistré au sein du DNS. Les nouveaux algorithmes sont alloués en utilisant la politique de consensus de l'IETF définie dans la RFC2434. Le nom d'algorithme HMAC-MD5.SIG-ALG.REG.INT ressemble à un FQDN pour des raisons historiques ; les noms d'algorithmes futurs seront vraisemblablement des noms simples (c'est-à-dire, d'un seul composant).

On attend de l'IANA qu'elle crée et entretienne un registre des "valeurs d'erreur de TSIG" à utiliser pour les valeurs de "Erreur" telles que définies au paragraphe 2.3. Les valeurs initiales devraient être celles définies au paragraphe 1.7. Les nouveaux codes d'erreur TSIG pour le champ Erreur TSIG sont alloués en utilisant la politique de consensus de l'IETF définie dans la RFC2434.

8. 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.

[RFC1321] R. Rivest, "Algorithme de résumé de message MD5", avril 1992. *(Information)*

[RFC1750] D. Eastlake, 3rd et autres, "Recommandations d'aléa pour la sécurité", décembre 1994. *(Info., remplacée par la RFC4086)*

[RFC2104] H. Krawczyk, M. Bellare et R. Canetti, "HMAC : Hachage de clés pour l'authentification de message", février 1997.

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

[RFC2136] P. Vixie, S. Thomson, Y. Rekhter et J. Bound, "Mises à jour dynamiques dans le système de noms de domaine (DNS UPDATE)", avril 1997.

[RFC2137] D. Eastlake 3rd, "Mise à jour dynamique sécurisée du système de noms de domaines", avril 1997. (*Obsolète, voir* RFC3007) *(MàJ* RFC1035) *(P.S.)*

[RFC2535] D. Eastlake, 3rd, "Extensions de sécurité du système des noms de domaines", mars 1999. (*Obsolète, voir* RFC4033, RFC4034, RFC4035) *(P.S.)*

[RFC2673] M. Crawford, "Étiquettes binaires dans le système des noms de domaine", août 1999. *(Exp. , MàJ par 3363, 3364)*

9. Adresse des auteurs

Paul Vixie
Internet Software Consortium
950 Charter Street
Redwood City, CA 94063
téléphone : +1 650 779 7001
mél : vixie@isc.org

Olafur Gudmundsson
NAI Labs
3060 Washington Road, Route 97
Glenwood, MD 21738
téléphone : +1 443 259 2389
mél : ogud@tislabs.com

Donald E. Eastlake 3rd
Motorola
140 Forest Avenue
Hudson, MA 01749 USA
téléphone : +1 508 261 5434
mél : dee3@torque.pothole.com

Brian Wellington
Nominum, Inc.
950 Charter Street
Redwood City, CA 94063
téléphone : +1 650 779 6022
mél : Brian.Wellington@nominum.com

10. Déclaration complète de droits de reproduction

Remerciement
Le financement de la fonction d'éditeur des RFC est actuellement assuré par la Internet Society.