rfc4035

Modifications de protocole pour les extensions de sécurité DNS

Les DNS Security Extensions (DNSSEC) sont une collection de nouveaux enregistrements et modifications de protocole qui ajoutent un authentification de l'origine des données et l'intégrité des données à DNS. Ce document définis les modifications du protocole DNSSEC.

Signature de zone

DNSSEC introdui le concept de zones signées. Une zone signée inclus une clé publique DNS (DNSKEY), un enregistrement de ressource de signature (RRSIG), Next Secure (NSEC), et optionnellement une délégation de signataire (DS). Une zone qui n'inclue pas ces enregistrements n'est pas une zone signée.

DNSSEC nécessite un changement dans la définition de l'enrgistrement de ressource CNAME, pour permettre aux RR RRSIG et NSEC d'apparaître avec le même nom propriétaire que le fait un RR CNAME.

DNSSEC spécifie le placement de 2 nouveaux type RR, RSEC et DS, qui peuvent être placés au niveau du parent (au point de délégation). C'est une exception à l'interdiction générale de placer des données dans la zone parent.

Inclure les RR DNSKEY dans une zone

Pour signer une zone, l'administrateur de zone génère une ou plusieurs paires de clés publique/privée et les utilise pour signer les RRset authoritatif dans la zone. Pour chaque clé privée utilisée pour créer les RR RRSIG dans une zone, la zone doit inclure un RR DNSKEY de zone contenant la clé publique correspondante. Une clé de zone RR DNSKEY doit avoir le bit de clé de zone du champ de flags RDATA mis. Les clés publiques associées avec d'autres opérations DNS peuvent être stockées dans des RR DNSKEY et ne sont pas marquées comme clé de zone et ne doivent pas être utilisées pour vérifier les RRSIG.

Si l'administrateur de zone prévois une zone signé pour être utilisable pour autre chose de la sécurité, l'apex de zone doit contenir au moins un RR DNSKEY pour agir comme point d'entrée sécurisé dans la zone. Ce point d'entrée sécurisé peut ainsi être utilisé comme cible d'une délégation sécurisé via un RR DS correspondant dans la zone parent.

Inclure les RR RRSIG dans une zone

Pour chaque RRset autoritatif dans une zone signée, il doit y avoir au moins un enregistrement RRSIG qui rencontre les exigences suivantes :

- Le nom propriétaire RRSIG est égal au nom propriétaire du RRset
- La classe RRSIG est la même que la classe du RRset
- Le champ Type Covered du RRSIG est égal au type du RRset
- Le champ Original TTL du RRSIG est égal au TTL du RRset
- Le champ Labels du RRSIG est égal au nombre de labels dans le nom propriétaire du RRset, sant compter le label root et le label le plus à gauche si c'est un wildcard
- Le champ Signer's Name du RRSIG est égale au nom de la zone contenant le RRset

- Les champs Algorithm, Signer's Name, et Key Tag identifient une clé de zone DNSKEY dans l'apex de la zone.

Le processus pour construire le RR RRSIG pour un RRset donné est décris dans la rfc4034. Un RRset peut avoir plusieurs RR RRSIG associé avec lui. Un RR RRSIG lui-même ne doit pas être signé, vu que signer un RR RRSIG n'ajoute pas de valeur et créé une boucle infinie dans le processus de signature.

Le RRset NS qui apparaît dans l'apex de la zone doit être signé, mais les RRset NS qui apparaîssent aux points de délégation ne doivent pas être signés. Les RRset Glue associés avec les délégation ne doivent pas être signés.

Il doit y avoir un RRSIG pour chaque RRset utilisant au moins une DNSKEY de chaque algorithm dans le RRset DNSKEY de l'apex de la zone. Le RRset DNSKEY apex lui-même doit être signé par chaque algorithme apparaissant dans le RRset DS localisé dans le parent déléguant.

Inclure les RR NSEC dans une zone

Chaque nom propriétaire dans la zone qui a une donnée autoritative ou un RRset NS point de délégation doit avoir un RR NSEC. Le format des RR NSEC et le processus pour le construire et donnée dans la rfc4034.

La valeur TTL pour un RR NSEC devrait être la même que le TTL minimum de la zone.

Un enregistrement NSEC (et son RRset RRSIG associé) ne doit pas être le seul RRset à un nom propriétaire particulier. C'est à dir, le processus de signature ne doit pas créer les RR NSEC ou RRSIG pour les nœuds de nom propriétaire qui n'est pas le nom propriétaire d'un RRset avant que la zone ne soit signée. La principale raison pour cela est un désire pour la consistance de l'espace de nom entre les versions signées et non signées de la même zone et un désire de réduire les risques de réponses inconsistantes dans les serveurs récursifs non sécurisés.

Le bitmap de tout RR NSEC dans une zone signée doit indiquer la présence de l'enregistrement NSEC lui-même et les enregistrement RRSIG correspondants.

La différence entre le jeu de noms propriétaire qui nécessite les enregistrements RRSIG et le jeu de noms propriétaire qui nécessitent les enregistrements NSEC est subtile et mérite d'être souligné. Les enregistrement RRSIG sont présent aux noms propriétaires de tous les RRset autoritatifs. Les enregistrements NSEC sont présents aux noms propriétaire et également aux noms propriétaires des délégations de la zone signée vers ses enfants. Ni NSEC ni RRSIG ne sont présents (dans la zone parente) aux noms propriétaire des RRset d'adresse glue. Noter, cependant, que cette distinction est généralement visible seulement durant le processus de signature de zone, vu que les RRset NSEC sont des données autoritatives et sont donc signées. Donc, tout nom propriétaire qui a un RRset NSEC aura des RR RRSIG également dans la zone signée.

Le bitmap pour le RR NSEC au point de délégation nécessite une attention spéciale. Les bits correspondants au RRset NS délégation et tout RRset pour lequel la zone parent an une donnée autoritative doit être mis ; les bits correspondants à un RRset non-NS pour lequel le parent n'est pas autoritatif doivent être effacés.

Inclure les RR DS dans une zone

Le RR DS établis les chaînes d'authentification entre les zones DNS. Un RRset DS devrait être présent à un point de délégation quand la zone enfant est signée. Le RRset DS peut contenir plusieurs enregistrement, chacun référençant une clé publique dans la zone enfant utilisée pour vérifier le RRSIG dans cette zone. Tous les RRset DS dans une zone doivent être signés, et les RRset DS ne doivent pas apparaître dans l'apex de la zone.

Un RR DS doit pointer vers un RR DNSKEY qui est présent dans le RRset DNSKEY de l'apex de l'enfant, et ce RRset doit être signé par la clé privée correspondante. Les RR DS qui échoue à ces conditios ne sont pas utilies pour la validation, mais parce que le RR DS et son RR DNSKEY sont dans des zones correspondantes, un erreur temporaire peut se produire.

Le TTL d'un RRset DS matche le TTL du RRset NS déléguant (la zone contenant le RRset DS).

La construction d'un RR DS nécessite la connaissance du RR DNSKEY correspondant dans la zone enfant, ce qui implique la communication entre les zones parent et enfant. Cette communication est un moyen opérationnel non couver par ce document.

Changement des RR CNAME

Si un RRset CNAME est présent au nom d'une zone signée, les RRset RRSIG et NESC sont requis à ce nom. Un RRset KEY à ce nom pour les mises à jours dynamique sécurisé est également fournis (rfc3007). D'autre types ne doivent pas être présent à ce nom.

C'est une modification de la définition du CNAME original de la rfc1034. La définition original du RR CNAME ne permet pas à d'autres type de cohexister avec un enregistrement CNAME, mais une zone signée exige les RR RRSIG et NSEC pour tout nom autoritatif. Pour résoudre ce conflit, cette spécification modifie la définition du RR CNAME pour lui permettre de cohexister vace les RR NSEC et RRSIG.

Type RR DNSSEC apparaissant aux coupures de zone

DNSSEC a introduit 2 nouveaux types de RR qui sont inhabituels par le fait qu'ils peuvent apparaître du côté du parent. Au niveau du point de délégation, les RR NSEC sont requit au nom propriétaire. Un RR DS peut également être présent si la zone à déléguer est signée et cherche à avoir une chaîne d'authentification dans la zone parent. C'est une exception de la spécification DNS original, qui status que seul les RRset NS peuvent apparaître dans le point de délégation du parent.

Cette spécification met à jours la spécification DNS pour permettre aux types RR NSEC et DS côté parent. Ces RRset sont autoritatif pour le parent quand ils apparaissent dans le point de délégation de la zone parent.

Service

Cette section décris le comportement des entités qui incluent les fonctions des serveurs de nom sécurisés. Dans beaucoup de cas de telles fonctions font partie d'un serveur de nom récursif sécurisé, mais un serveur de nom autoritatif sécurisé possède certaines de ces exigences.

Un serveur de nom sécurisé doit supporter EDNS0 (rfc2671), une taille de message d'au moins 1220 octets, et devrait supporter une taille de message de 4000 octets. Vu que les paquets IPv6 peuvent seulement être fragmentés par l'hôte source, un serveur de nom sécurisé devrait prendre en considération les étapes pour s'assurer que les datagrammes UDP qu'il transmet sur IPv6 sont fragmentés, si nécessaire, au MTU IPv6 minimum.

Un serveur de nom sécurisé qui reçois une requête DNS qui n'inclus pas l'option EDNS pseudo-RR ou sans le bit DO doit traite les RR RRSIG, DNSKEY et NSEC comme si c'était un RRset sans traitement additionnel. Parce que le type RR DS a la propriété particulière de n'exister que dans la zone parente, les RR DS exigent toujours un traitement spécial.

Les serveurs de nom sécurisés qui reçoivent des requêtes explicite pour les types RR de sécurité qui matchent le contenu de plus d'une zone qu'il déssert (par exemple, les RR NSEC, et RRSIG avant et après un point de délégation œu le sereur est autoritatif pour les 2 zones) devraient de comporter comme consistant. Tant que la réponse est toujours consistante pour chaque requête au serveur de nom, le serveur peut retourner au choix :

- Les RRsets au dessus de la délégation
- Les RRsets au dessous de la délégation
- Les 2
- Une section réponse vide

- Une autre réponse
- Une erreur

DNSSEC alloue 2 nouveaux bits dans l'en-tête de message DNS : CD (Checking Disabled) et AD (Authentic Data). Le bit CD est contrôlé par les résolveurs. Un serveur de nom sécurisé doit copier le bit CD de la requête dans la réponse correspondante. Le bit AD est contrôlé par les serveurs de noms, et un serveur de nom sécurisé doit ignorer le bit AD dans les requêtes.

Un serveur de nom sécurisé qui synthétise les RR CNAME depuis les RR DNAME (rfc2672) ne devrait pas générer de signatures pour les RR CNAME synthétisés.

Serveur de nom autoritatif

Une fois une requête reçue qui a le bit DO de l'option EDNS(0) pseudo-RR, un serveur de nom autoritatif sécurisé pour une zone signée doit inclure les RR RRSIG, NSEC et DS, en accord avec les règles suivantes :

- Les RR RRSIG qui peuvent être utilisés pour authentifier une réponse doivent être inclus dans la réponse
- Les RR NSEC qui peuvent être utilisé pour fournir un refus d'existance authentifié doivent être inclus dans la réponse
- Soit un RRset DS ou un RR NSEC prouvant qu'aucun RR DS n'existe doit être inclus dans les référrants

Ces règles s'appliquent seulement aux réponses où les sémantiques véhiculent des informations sur la présence ou l'absence d'enregistrement de ressource. C'est à dire, Ces règles ne sont pas prévues pour pour exclure les réponses tels que RCODE 4 (Not Implemented) ou RCODE 5 (Refused).

DNSSEC ne change pas le protocole de transfert de zone.

Inclure les RR RRSIG dans une réponse

En répondant à une requête qui a le bit DO mis, un serveur de nom devrait tenter d'envoyer les RR RRSIG qu'un résolveur peut utiliser pour authentifier les RRset dans la réponse. Un serveur de nom devrait tenter de conserver le RRset et ses RRSIG ensemble dans une réponse. L'ajout des RR RRSIG dans une réponse est sujet aux règles suivantes :

- En plaçant un RRset signé dans une section Answer, le serveur de nom doit également placer ses RR RRSEG dans la section Answer. Les RR RRISG ont une priorité supérieur pour l'inclusion que tout autre RRset qui peuvent être inclus. Si l'espace ne permet pas d'inclure ces RR RRSIG, le serveur de nom doit mettre le bit TC.
- En plaçant un RRset signé dans la section Authority, le serveur de nom doit également placer ses RR RRSIG dans la section Authority. Les RR RRSIG ont une priorité supérieur pour l'inclusion que tout autre RRset qui peuvent être inclus. Si l'espace ne permet pas d'inclure ces RR RRSIG, le serveur de nom doit mettre le bit TC.
- En plaçant un RRset signé dans la section Additional, le serveur de nom doit également placer ses RR RRSIG dans la section Additional. Si l'espace ne permet pas d'inclure le RRset et ses RR RRSIG associés, le serveur de nom peut retenir le RRset et enlever les RR RRSIG. Si cela se produit, le serveur de nom ne doit pas définir le bit TC.

Inclure les RR DNSKEY dans une réponse

En répondant à une requête qui a le bit DO mis et que demande les RR SOA et NS à l'apex d'une zone signée, un serveur de nom autoritatif pour cette zone peut retourner le RRset DNSKEY de l'apex dans la section Additional. Dans cette situation, le RRset DNSKEY et les RR RRSIG associés ont une priorité inférieur que d'autres informations à placer dans la section additionnelle. Le server de nom ne devrait pas inclure le RRset DNSKEY sauf s'il y a suffisamment de place dans la réponse pour le RRset DNSKEY et les RR RRSIG associés. S'il n'y a pas assez de place, le server de nom doit les omettre et ne doit pas inclure le bit TC uniquement pour ces RR.

Inclure les RR NSEC dans une réponse

En répondant à une requête qui a le bit DO mis, un serveur de nom autoritatif pour une zone signée doit inclure les RR NSEC dans chacun de ces cas :

No Data : La zone contient des RRset qui matche exactement <SNAME, SCLASS> mais ne contient pas de RRset qui match exactement <SNAME, SCLASS, STYPE>.

Name Error : La zone ne contient pas de RRset qui match <SNAME, SCLASS>, soit exactement, soit via l'expansion wildcard

Wildcard Answer : La zone ne contient pas le RRset qui match exactement <SNAME, SCLASS> mais contient un RRset qui match <SNAME, SCLASS, STYPE> via l'expansion wildcard

Wildcard No Data : La zone ne contient pas de RRset qui match exactement <SNAME, SCLASS> et contient un ou plusieurs RRset que match <SNAME, SCLASS> via l'expansion wildcard, mais ne contient pas de RRset qui match <SNAME, SCLASS, STYPE> via l'expansion wildcard.

Dans chacun de ces cas, le serveur de nom inclus les RR NSEC dans la réponse pour prouver qu'un match exact pour <SNAME, SCLASS, STYPE> n'était pas présent dans la zone et que la réponse que le serveur retourne les données dans la zone est correct.

Inclure les RR NSEC : No Data Response

Si la zone contient des RRset matchant <SNAME, SCLASS> mais ne contient pas de RRset matchant <SNAME, SCLASS, STYPE>, le serveur de nom doit inclure le RR NSEC pour <SNAME, SCLASS>, puis le serveur de nom doit inclure le RR NSEC pour <SNAME, SCLASS> avec les RR RRSIG associés dans la section Authority de la réponse. Si l'espace ne permet pas d'inclure le RR NSEC ou les RR RRSIG associés, le serveur de nom doit mettre le bit TC.

Vu que le nom recherché existe, l'expansion wildcard ne s'applique par à cette requête, et un simple RR NSEC suffit à prouver que le type RR demandé n'existe pas.

Inclure les RR NSEC : Name Error Response

Si la zone ne contient pas de RRset matchant <SNAME, SCLASS> soit exactement soit via l'expansion wildcard, le serveur doit incluse les RR NSEC suivants dans la section Authority, avec les RR RRSIG associés :

- Un RR NSEC prouvant qu'il n'y a pas de match exace pour <SNAME, SCLASS>
- Un RR NSEC prouvant que la zone ne contient pas de RRset qui match <SNAME, SCLASS> via l'expansion wildcard

Dans certains cas, un simple RR NSEC peut prouver ces 2 points. Si c'est le cas, le serveur devrait seulement inclure le RR NSEC et les RR RRSIG une seule fois dans la section Authority

Si l'espace ne le permet pas, le serveur doit mettre le bit TC.

Les noms propriétaires de ces RR NSEC et RRSIG ne sont pas sujet à l'expansion wildcard quand ces RR sont inclus dans la section Authority de la réponse.

Noter que cette forme de réponse inclus les cas dans lesquels SNAME correspond à un nom non-terminal vide dans la zone (un nom qui n'est pas le nom propriétaire pour un RRset mais qui est le nom parent d'un ou plusieurs RRset).

Inclure les RR NSEC : Wildcard Answer Response

Si la zone ne contient pas de RRset qui matche exactement <SNAME, SCLASS> mais contient un RRset qui matche <SNAME, SCLASS, STYPE> via l'expansion wildcard, le serveur de nom doit inclure la réponse étendue et les RR RRSIG étendus correspondants dans la section Answer et doit inclure dans la section Authority un RR NSEC et les RR RRSIG associés prouvant que la zone ne contient pas de match à <SNAME, SCLASS>. Si l'espace ne permet pas d'inclure les RR NSEC et RRSIG, le serveur doit mettre le bit TC.

Inclure les RR NSEC : Wildcard No Data Response

Ce cas est une combinaison des précédents cas. La zone ne contient pas de match exacte pour <SNAME, SCLASS>, et bien que la zen contient des RRset qui match <SNAME, SCLASS> via l'expansion wildcard, aucun RRset ne match STYPE. Le serveur de nom doit inclure les RR NSEC suivants dans la section Authority, avec les RR RRSIG associés :

- Un RR NSEC prouvant qu'il n'y a pas de RRset correspondant à STYPE au nom propriétaire wildcard qui match <SNAME, SCLASS>
- Un RR NSEC prouvant qu'il n'y a pas de RRset dans la zone qui aurait matché pour <SNAME, SCLASS>

Dans certains cas, un simple RR NSEC peut prouver les 2. Les noms propriétaire de ces RR NSEC et RRSIG ne sont pas sujet à l'expansion wildcard quand ces RR sont inclus dans la section Authority de la réponse. Si l'espace ne permet pas d'inclure les RR NSEC et RRSIG, le serveur de nom doit mettre le bit TC.

Trouver les bon RR NSEC

Comme expliqué plus haut, il y a de nombreuses situations dans lesquelles un serveur de nom doit localiser un RR NSEC qui prouve qu'aucun RRset correspondant à un SNAME n'existe. Localiste un tel RR NSEC dans une zone autoritative est relativement simple, du moins dans le concept. La suite assume que le serveur de nom est autoritatif pour la zone. L'algorithme ci-dessous est écrit pour clareté, non pour l'efficacité.

Pour trouver les NSEC qui prouve qu'aucun RRset ne match le nom N dans la zone Z, constuire une séquence, S, consistant des noms propriétaire de tout RRset dans Z, trié dans l'ordre canonique, sans noms dupliqué. Trouver le nom M qui aurait précédé immédiatement N dans S si un RRset avec le nom N existait. M est le nom propriétaire du RR NSEC qui prouve qu'aucun RRset n'existe avec le nom N.

L'algorithme pour trouver le RR NSEC qui prouve qu'un nom donné n'est pas couvert par un wildcard applicable est similaire mais nécessite une étape supplémentaire. Plus précisemment, l'algorithme pour trouver le NSEC prouvant qu'aucun RRsec n'existe avec le nom wildcard applicable est précisemment le même que l'algorithme pour trouver le RR NSEC qui prouve que les RRset avec un nom propriétaire n'existe pas. La partie manquante est une méthode pour déterminer le nom du wildcard applicable non-existant. En pratique, c'est facile, parce que le serveur de nom autoritatif à déjà vérifié la présence de ce nom wildcard dans l'étape (1)(c) de l'algorithme de recherche décrit dans la rfc1034.

Inclure les RR DS dans une réponse

En répondant à une requête qui a le bit DO mis, un serveur de nom autoritatif retournant un référant inclus les données DNSSEC avec le RRset NS.

Si un RRset DS est présent au point de délégation, le serveur de nom doit retourner le RR DS et ses RR RRSIG associés dans la section Authority avec le RRset NS.

Si aucun RRset DS n'est présent au point de délégation, le serveur de nom doit retourner le RR NSEC qui prouve que le RRset DS n'est pas présent et les RR RRSIG associés du RR NSEC avec les RRset NS. Le serveur de nom doit placer le RRset NS avant les RRset NSEC et les RR RRSIG associés.

En incluant ces RR DS, NSEC, et RRSIG, la taille de message augmente et peut causer l'omission de tous les RR glue. Si l'espace ne permet pas d'inclure les RRset DS ou NSEC et RRSIG, le serveur de nom doit mettre le bit TC.

Répondre aux requêtes pour les RR DS

Le type RR DS n'est pas courant par le fait qu'il apparaît seulement dans la zone parente. Par exemple, le RRset DS pour la délégation de "foo.example" est stockée dans la zone "example". Cela nécessite un traitement spéciale pour les serveurs de nom et les résolveurs, vu que le serveur de nom pour la zone enfant est autoritative pour le nom à la coupure de zone par les règles DNS normales mais la zone enfant ne contient pas le RRset DS.

Un résolveur sécurisé envoie des requêtes à la zone parent en recherchant un RR DS au point de délégation. Cependant, des règles spéciales sont nécessaires pour éviter la confusion des résolveurs non sécurisés qui peuvent traiter une telle requête (par exemple, dans une configuration réseau qui force un résolveur sécurisé à canaliser ses requêtes via un serveur de nom récursif non sécurisé). Le reste de cette section décris comment un serveur de nom sécurisé traite les requêtes DS pour éviter ce problème.

Le besoin pour un traitement spécial par un serveur de nom sécurisé survient seulement quand toutes les conditions suivantes sont rencontrées :

- Le serveur de nom a reçu une requête pour le RRset DS à la coupure de zone
- Le serveur de nom est autoritaire pour la zone enfant
- Le serveur de nom n'est pas autoritatif pour la zone parent
- Le serveur de nom n'offre pas de récursion

Dans tous les autres cas, le serveur de nom a soit une manière d'obtenir le RRset DS soit ne s'attend pas à avoir le RRset DS même si les règles de traitement pré-DNSSEC, donc le serveur de nom peut retourner soit le RRset DS ou une erreur en accord avec les règles de traitement normal.

Si toutes les conditions ci-dessus sont rencontrées, cependant, le serveur de nom est autoritatif pour SNAME mais ne peut pas fournir le RRset demandé. Dans ce cas, le serveur de nom doit retourner une réponse "no data" montrant que le RRset DS n'existe pas dans l'apex de la zone enfant.

Répondre aux requêtes pour le type AXFR ou IXFR

DNSSEC ne change par le processus de transfert de zone. Une zone signée contient les RR RRSIG, DNSKEY, NSEC, et DS, mais ces enregistrement n'ont pas de signification spéciale dans les opérations de transfert de zone.

Un serveur de nom autoritatif n'est pas tenu de vérifier qu'une zene est proprement signée avent d'envoyer ou d'accepter un transfert de zone. Cependant, un serveur de nom autoritatif peut choisir de rejeter tout le transfert de zone si la zone ne répond pas à toutes les exigences décrites dans la section Signature de zone. L'objectif principal d'un transfert de zone est de s'assurer que tous les serveurs de nom autoritatifs ont une copie identique de la zone. Un serveur de nom autoritatif qui choisis d'effectuer sa propre validation de zone ne doit pas rejeter sélectivement certains RR et en accepter d'autres.

Les RRset DS apparaîssent seulement coté parent d'une coupure de zone et sont des données autoritatives dans la zone parent. Comme avec tout autre RRset autoritatif, le RRset DS doit être inclus dans le transfert de zone dans lequel le RRset est une donnée autoritative. Dans la cas du RRset DS, c'est la zone parent.

Les RR NSEC apparaissent dans les zones parent et enfant à la coupure de zone et sont autoritatifs dans ces 2 zones. Les RR NSEC à la coupure de zone parent et enfant ne sont jamais identique, vu que le RR NSEC dans l'apex de la zone enfant indiquent toujours la présence du RR SOA de la zone enfant alors que le RR NSEC du parent ne l'indique pas. Comme avec tout autre RR autoritatif, les RR NSEC doivent être inclus dans les transferts de zone dans lequels ils sont des données autoritatives. Le RR NSEC parent doit être inclus dans le transfert de zone de la zone parent, et le NSEC de l'apex de la zone enfant doit être inclus dans le transfert de la zone enfant.

Les RR RRSIG apparaissent dans les zones parent et enfant à la coupure de zone et sont autoritatifs dans la zone contenant contenant le RRset autoritatif pour lequel le RR RRSIG fournis la signature. C'est à dire, le RR RRSIG pour un RRset DS ou un RR NSEC parent à la coupure de zone sera autoritatif dans la zone parent, et le RRSIG pour un RRset dans l'apex de la zone enfant sera autoritatif dans la zone enfant. Les RR RRSIG parent et enfant à la coupure de jone ne sont jamais identique. Comme d'autres RR autoritatif, les RR RRSIG doivent être inclus dans le transfert de zone dans lequel ils sont des données autoritatives.

bits AD et CD dans une réponse autoritative

Les bits CD et AD sont conçus pour être utilisés dans une communication entre des résolveurs sécurisés et des serveurs de nom récursifs sécurisés. Ces bits ne sont pas significatif pour le traitement par les serveurs de nom autoritatifs.

Un serveur de nom sécurisé ne valide pas la signature pour une donnée autoritative durant le traitement de la requête, même quand le bit CD est effacé. Un serveur de nom sécurisé devrait effacer le bits CD en créant une réponse autoritative.

Un serveur de nom sécurisé ne doit pas mettre le bit AD dans une réponse à moins que le serveur considère que tous les RRset dans les section Answer et Authority de la réponse sont authentiques. La stratégie locale d'un serveur de nom sécurisé peut considérer les données d'une zone autoritative comme authentique sans autre validation. Cependant, le serveur de nom ne doit pas le faire à moins qu'il n'obtienne la zone autoritative via un moyen sécurisé (tel qu'un mécanisme de transfert de zone sécurisé) et ne doit pas le faire sauf si ce comportement a été configuré explicitement.

Un serveur de nom sécurisé qui supporte la récursion doit suivre les règles suivantes pour les bits CD et AD donnés ci-dessous en générant un réponse qui implique des données obtenus via la récursion.

Servers de nom récursifs

Comme indiqué dans la rfc4033, un serveur de nom récursif sécurisé est une entité qui agit dans les rôles de serveur de nom sécurisé et de résolveur sécurisé. Cette section utilise les termes "partie serveur de nom" et "partie résolveur" pour référrer au code dans un serveur de nom qui implémente ces 2 rôles.

La partie résolveur suis les règles usuelles pour le caching et le caching négatif qui s'applique à tout résolveur sécurisé

Le bit DO

La partie résolveur doit mettre le bit DO en envoyant les requêtes, sans regarder l'état du bit DO dans la requête initiale reçue par la partie serveur de nom. Si le bit DO dans une requête initiale n'est pas mise, la partie serveur de nom doit enlever tous RR DNSSEC authentifiant de la réponse mais ne doit pas enlever les type RR DNSSEC que la requête initiale a demandé explicitement.

Le bit CD

Le bit CD existe pour permettre à un résolveur sécurisé de désactiver la validation de la signature dans le traitement du serveur de nom d'une requête particulière.

La partie serveur de nom doit copier le paramètre du bit CD de la requête à la réponse correspondante.

La partie serveur de nom doit passer l'état du bit CD à la partie résolveur avec le reste de la requête initiale, pour que la partie résolveur sache s'il est nécessaire de vérifier la réponse qu'il retourne à la partie serveur de nom. Si le bit CD est mis, il indique que le résolveur

d'origine souhaite effectuer une authentification via sa stratégie locale. Donc, la partie résolveur n'a pas besoin d'effectuer d'authentification dans les RRset dans la réponse. Quand le bit CD est mis, le serveur de nom récursif devrait, si possible, retourner la donnée demandée au resolveur d'origine, même si la stratégie locale du serveur aurait rejeté les enregistrements en question. C'est à dire, en définissant le bit CD, le résolveur d'origine a indiqué qu'il prend la responsabilité pour effectuer sa propre authentification, et le serveur de nom n'interfers pas.

Si la partie résolveur implémente un cache BAD et la partie serveur de nom reçois une requête qui matche une entrée dans ce cache, La réponse de la partie serveur dépend de l'état du bit CD dans la requête originale. Si le bit CD estmis, la partie serveur de nom devrait retourner la données depuis le cache ; si le bit CD n'est pas mis, la partie serveur de nom doit retourner le RCODE 2 (server failure)

L'intention de cette règle est de fournir les données brute aux clients capable d'effectuer eux-même la vérification de la signature tout en protégeant le client qui dépent de cette vérification.

Le bit AD

La partie serveur de nom ne doit pas mettre le bit AD dans une réponse à moins que le serveur considère tous les RRset dans les section Answer et Authority de la réponse comme étant authentiques. La partie serveur de nom devrait mettre le bit AD si et seulement si la partie résolveur considère tous les RRset dans les serveur Answer et Authority comme étant authentiques. La partie résolveur doit suivre la procédure décrite dans la section "Authentifier les réponses DNS" pour déterminer si les RR en question sont authentiques. Cependant, pour compatibilité, un serveur de nom récursif peut mettre le bit AD quand une réponse incluse des RR CNAME non signés si ces RR CNAME démontrent avoir été synthétisés depuis un RR DNAME authentique qui est également inclus dans la réponse en accord avec les règles de synthèse de la rfc2672.

Résolution

Cette section décrit le comportement des entités qui incluent des fonctions de résolution sécurisé. Dans la plupart des cas de telles fonctions font partie d'un serveur de nom récursif sécurisé, mais un résolveur sécurisé a de nombreuses exigences similaires. Les fonctions spécifique aux serveurs de nom récursif sécurisés sont décrits dans la section précédente.

Support EDNS

Un résolveur sécurisé doit inclure un OPT pseudo-RR EDNS avec le bit DO mis en envoyant les requêtes

Un résolveur sécurisé doit supporter une taille de message d'au moins 1220 octets, devrait supporter une taille de message de 4000 octets, et doit utiliser le champ sender,' UDP payload size dans le pseudo-RR pour annoncer la taile de message qu'il est prêt à accepter. Un couche IP du résolveur sécurisé doit gérer les paquets UDP fragmentés correctement sans regarder si ces fragments sont reçus via IPv4 ou IPv6.

Support de vérification de signature

Un résolveur sécurisé doit supporter les mécanismes de vérification de signature et devrait les appliquer à chaque réponse reçue, excepté quand :

- Le résolveur sécurisé fait partie d'un serveur de nom récursif, et la réponse est le résultat d'une récursion à la demande d'une requête reçue avec le bit CD mis ;
- La réponse est le résultat d'une requête générée directement via une interface d'application qui a instruit le résolveur de ne pas effectuer de validation pour cette requête ;

- La validation pour cette requête a été désactivée par stratégie locale.

Le support d'un résolveur sécurisé pour la vérification de signature doit inclure le support pour la vérification des noms propriétaires wildcard.

Les résolveurs sécurisés peuvent requêter les RR de sécurité manquant en tentant d'effectuer une validation ; les implémentations qui choisissent de le faire doit être prêt à recevoir une réponse qui n'est pas suffisante pour valider la réponse originale. Par exemple, une mise à jours de zone peut avoir changé ou supprimé l'information désirée entre les requêtes original et les requêtes suivantes.

En tentant de récupérer les RR NSEC manquant qui résident dans le parent, un résolveur en mode itératif doit requêter les serveurs de nom pour la zone parent, pas la zone enfant.

En tentant de récupérer un DS manquant, un résolveur en mode itératif doit requêter les serveurs de nom pour la zone parent. Comme vu plus haut, les serveurs de nom doivent appliquer des traitements spéciaux pour gérer le RR DS, et dans certaines situations le résolveur peut également nécessiter d'appliquer les règles spéciales pour localiser les serveurs de nom pour la zone parent si le résolveur n'a pas le RRset NS du parent. Pour localiser le RRset NS parent, le résolveur peut commencer avec le nom de délégation, enlever le label le plus à gauche, et requêter un RRset NS par ce nom. Si aucun RRset NS n'est présent à ce nom, le résolveur enlève le label le plus à gauche et retente la requête pour ce nom, en répétant le processus jusqu'à ce qu'il trouve le RRset NS ou n'ai plus de label.

Déterminer le status de sécurité des données

Un résolveur doit être capable de déterminer s'il doit s'attendre à un RRset particulier signé. Plus précisemment, un résolveur sécurisé doit être capable de faire la distinction entre 4 cas :

Sécure : Un RRset pour lequel le résolveur est capable de construire une chaîne de RR DNSKEY et DS signés depuis une ancre de confiance au RRset. Dans ce cas, le RRset devrait être signé et est sujet à validation de signature.

Insecure : Un RRset pour lequel le résolveur sait qu'il n'a pas de chaîne de RR DNSKEY et DS signé. Cela se produit quand le RRset cible est dans une zone non signée. Dans ce cas le résolveur ne peut pas vérifier la signature.

Bogus : Un RRset pour lequel le résolveur estime qu'il est en mesure d'établir une chaîne de confiance, mais n'est pas capable de le faire, soit en raison de signature qu'il ne parvient par à valider, soit à cause de données manquantes que les RR DNSSEC indiquent qu'elles devraient être présentes. Ce cas peut indiquer une attaque ou une erreur de configuration.

Indeterminate : Un RRset pour lequel le résolveur n'est pas capable de déterminer si le RRset est signé, vu que le résolveur n'est pas capable d'obtenir les RR DNSSEC nécessaire. Cela peut se produire quand le résolveur sécurisé n'est pas capable de contacter les serveurs de nom pour les zones concernées.

Ancres de confiance configurés

Un résolveur sécurisé doit être capable d'être configuré avec au moin une clé publique ou ur RR DS et devrait être capable d'être configuré avec plusieurs clé ou RR DS. Vu qu'un résolveur sécurisé n'est pas capable de valider les signatures sans ancre de confiance, le résolveur devrait avoir un mécanisme robuste pour obtenir de telles clé quand il démarre ; par exemple un stockage non-volatile (disque dur) ou une configuration réseau local de confiance. Noter que les ancres de confiance couvrent également les clé qui sont mis à jours de manière sécurisé, qui peut être un support physique, un protocole d'échange de clé, ou d'autres moyens tiers.

Cacher les réponses

Un résolveur sécurisé devrait mettre en cache chaque réponse comme simple entrée atomique contenant toute la réponse, incluant le RRset nomé et tout RR DNSSEC associé. Le résolveur devrait supprimer l'entrée quand un des RR qu'il contient expire. Dans beaucoup de cas le l'index de cache approprié pour l'entrée atomique est le triplet <QNAME, QTYPE, QCLASS>, mais pour les cas décris dans la section "inclure les RR NSEC : Name Error Response", l'index sera <QNAME, QCLASS>.

La raison de ces recommandations est que, entre la requête initiale et l'expiration des données du cache, les données autoritatives peuvent avoir changé. Il y a 2 situation pour lesquel c'est important :

1. En utilisant l'enregistrement RRSIG, il est possible de déduire qu'une réponse a été synthétisée depuis un wildcard. Un serveur de nom récursif pourrait stocker cette donnée wildcard et l'utiliser pour générer des réponses positives aux requêtes autre que le nom pour lequel la réponse originale a été reçue.
2. Les RR NSEC reçues pour prouver la non-existance d'un nom pourrait être réutilisée par un résolveur pour prouver la non-existence d'un nom dans la plage de noms.

En théorie, un résolveur pourrait utiliser les wildcard ou les RR NSEC pour générer des réponses positives et négatives (respectivement) jusqu'à ce que le TTL ou les signatures dans les enregistrement n'expirent. Cependant, il semble prudent pour les résolveurs d'éviter de bloquer les nouvelles données autoritatives ou de synthétiser de nouvelles donnés par eux même. Les résolveurs qui suivent cette recommandation auront une vue plus consistante de l'espace de nom.

Gérer les bits CD et AD

Un résolveur sécurisé peut mettre le bit CD d'une requête pour indiquer que le résolveur prend la résponsabilité d'effectuer toute authentification que sa stratégie locale exige sur les RRsets dans la réponse.

Un résolveur doit effacer le bit AD en créant une requête pour se protéger contre les serveurs de nom buggés qui copient bêtement les en-têtes qu'ils ne comprennent pas de la requête dans la résponse.

Un résolveur doit ignorer la signification des bits CD et AD dans une réponse sauf si la réponse a été obtenue en utilisant un canal sécurisé ou le résolveur a été configuré spécifiquement pour le faire.

Cache des données BAD

Bien que de nombreuses erreurs de validation sont transitoires, certaines sont persistantes, tel que les erreurs administratives. Vu que redemander n'aide pas dans ces cas, les résolveurs validateurs peuvent générer un quantité significative de trafique DNS non-nécessaire en mettant en cache les signatures invalides, avec quelques réstrictions.

Conceptuellement, cacher de telles données est similaire aux caching négatif (rfc2308), excepté qu'au lieu de cacher une réponse invalide, le résolveur cache le fait qu'une réponse particulière à échoué la validation. Ce document réfèrre au cache de telles données au cache BAD.

Les résolveurs qui implémentent un cache BAD doit effectuer certaines étapes pour éviter que le cache soit utilisé comme amplificateur d'attaque DOS, particulièrement :

- Vu que les RRset qui échouent la validation n'ont pas de TTL de confiance, l'implémentation doit leur assigner un TTL. Ce TTL devrait être petit, pour mitiger l'effet de cache de résultat d'une attaque.
- Pour éviter de cacher des erreurs de validation aléatoires (qui peuvent être le résultat d'une attaque), les résolveurs devraient suivre les requêtes qui résultent en erreurs de validation de devraient seulement répondre depuuis le cache BAD après que le nombre de fois que les réponse aux requêtes pour ce <QNAME, QTYPE, TCLASS> particulier a échoué la validation a excédé un valeur seuil.

Les résolveurs ne doivent pas retourner les RRset depuis le cache BAD sauf si le résolveur n'a pas à valider les signature des RRset en question.

CNAME synthétisés

Un résolveur sécurisé validateur doit traiter la signature d'un RR DNAME signé valide comme couvrant égaleent les RR CNAME non-signés qui pourraient avoir été synthétisés depuis le RR DNAME, tel que décris dans la rfc2672, au moins dans la mesure de ne pas rejeter une réponse parce qu'elle contient seulement de tels RR CNAME.

Résolveurs stub

Un résolveur stub sécurisé doit supporter les types de RR DNSSEC, au moins pour éviter de mal gérer les réponses simplement parce qu'elles contiennent des RR DNSSEC.

Gérer les bit DO

Un résolveur stub sécurisé non-validateur peut inclure les RR DNSSEC retournés par un serveur de nom récursif sécurisé comme partie des données que le stub resolver envoie à l'application, mais ce n'est pas obligatoire. Un stub resolver qui le fait doit mettre le bit DO pour recevoir les RR DNSSEC depuis le serveur de nom récursif.

Un stub resolver sécurisé validateur doit mettre le bit DO, parce que sinon il ne reçoit pas les RR DNSSEC qu'il a besoin pour la validation de la signature.

Gérer le bit CD

Un stub resolver sécurisé non-validateur ne devrait pas mettre le bit CD en envoyant les requêtes sauf s'il est demandé par l'application, vu que par définition, un stub resolver non-validateur dépend du serveur de nom récursif sécurisé qui effectue la validation pour lui.

Un résolveur stub sécurisé validateur devrait mettre le bit CD, parce que sinon le serveur de nom récursif va répondre à la requête en utilisant la stratégie locale du serveur de nom, qui peut empêcher le stub resolver de reçevoir des données qui seraient acceptables pour la stratégie local du stub resolver.

Gérer le bit AD

Un stub resolver sécurisé non-validateur peut choisir d'examiner le bit AD dans les réponses qu'il reçoit pout déterminer si le serveur de nom récursif sécurisé qui envoie la réponse prétend avoir cryptographiquement vérifié les données dans les section Answer et Authority de la réponse. Noter, cependant, que les réponse reçues par un stub resolver sécurisé dépend de la stratégie locale du serveur de nom sécurisé. Un stub resolver ne doit pas placer de confiace dans la prétendue validation de signature effectuée, exceptée quand le stub resolver sécurisé a obtenu les données en question depuis un serveur de nom récursif sécurisé de confiance via un canal sécurisé.

Un résolveur stub validateur ne doit pas examiner le bit AD dans les messages de réponse, vu que par définition, le stub resolver effectue sa propre validation de signature.

Authentifier les réponses DNS

Pour utiliser les RR DNSSEC pour d'authentification, un résolveur doit connaître au moins un RR DNSKEY ou DS authentifié. Le processus pour obtenir et authentifier cet ancre de confiance initial est fait via un mécanisme externe. Le reste de cette section assume que le résolveur a obtenu le jeu initial d'ancres de confiance.

Un RR DNSKEY initial peut être utilisé pour authentifier un RRset DNSKEY dans l'apex de la zone. Pour authentifier un RRset DNSKEY apex en utilisant une clé initiale, le résolveur doit :

1. Vérifier que le RR DNSKEY initial apparaît dans le RRset DNSKEY apex, et que le RR DNSKEY a le Zone Key Flag mis.
2. Vérifier qu'il y a un RR RRSIG qui couvre le RRset DNSKEY apex, et que la combinaison du RR RRSIG et du RR DNSKEY authentifient le RRset DNSKEY. Le processus pour utiliser un RR RRSIG pour authentifier un RRset est décris plus bas.

Une fois que le résolveur a authentifié le RRset DNSKEY apex en utilisant un RR DNSKEY initial, les délégation depuis cette zone peuvent être authentifiés en utilisant les RR DS. Celà permet à un résolveur de démarrer depuis une clé initiale et d'utiliser les RRset DS pour traiter récursivement l'arborescence DNS, en obtenant les autres RRset DNSKEY apex. Si le résolveur était configuré avec un RR DNSKEY root, et si chaque délégation a un RR DS associé avec lui, le résolveur peut obtenir et valider tout RRset DNSKEY apex. Le processus d'utilisation les RR DS pour authentifier les référrants est décris plus bas.

Quand un résolveur indique le support pour DNSSEC (en définissant le bit DO), un serveur de nom sécurisé devrait tenter de fournir les RRset DNSSEC nécessaires dans une réponse. Cependant, un résolveur sécurisé peut reçevoir une réponse dans laquelle il manque des RR DNSSEC. Un résolveur devrait attendre les informations d'authentification des zones signées. Un résolveur devrait croire qu'une zone est signée si le résolveur a été configuré avec des informations de clé publique pour la zone, ou si le parent de la zone est signée, et contient un RRset DS.

Considérations spéciales pour les îlots de sécurité

Les îlots de sécurité sont les zones signées pour lequelles il n'est pas possible de construire une chaîne d'authentification vers la zone depuis son parent. Valider les signature dans un îlot de sécurité nécessite que le validateur ait un moyen d'obtenir une clé de zone authentifiée initiale pour l'îlot. Si un validateur ne peut pas obtenir une telle clé, il devrait opérer comme si les zones étaient non-signées.

Tous les processus normaux pour valider les réponse s'appliquent aux îlots de sécurité. La seul différence entre une validation normale et la validation d'une un îlot de sécurité et la manière dont le validateur obtient une ancre de confiance pour la chaîne d'authentification.

Authentifier les référrants

Une fois le RRset DNSKEY apex pour une zone parent signée est authentifiée, les RRset DS peuvent être utilisés pour authentifier la délégation pour une zone enfant signée. Un RR DS identifie un RR DNSKEY dans le RRset DNSKEY apex de la zone enfant. Un RR DS identifie un RR DNSKEY dans le RRset DNSKEY dans l'apex de la zone enfant. L'utilisation d'un algorithme de hashage fort s'assure qu'il est impossible pour un attaquant de générer un RR DNSKEY qui matche le hash. Donc, authentifier le hash permet au résolveur d'authentifier le RR DNSKEY. Le résolveur peut ainsi utiliser ce RR DNSKEY pour authentifier tout le RRset DNSKEY apex enfant.

En donnant un RR DS pour une délégatio, le RRset DNSKEY apex de la zone enfant peut être authentifié si :

- Le RR DS a été authentifié en utilisant un RR DNSKEY dans le RRset DNSKEY apex du parent.
- Algorithm et Key Tag dans le RR DS match les champs Algorithm et Key Tag d'un RR DNSKEY dans le RRset DNSKEY apex de la zone enfant, et, quand le nom propriétaire du RR DNSKEY et RDATA sont hashés en utilisant l' algorithme de hashage spécifié dans le champ Digest Type dans le RR DS, les valeurs de hash correspondent.
- Le RR DNSKEY correspondant dans la zone enfant a le bit Zone Flag mis, la clé privée correspondante a signé le RRset DNSKEY apex de la zone, et le RR RRSIG résultant authentifie le RRset DNSKEY apex de la zone enfant.

Si le référant de la zone parent ne contient pas un RRset DS, la réponse devrait inclure un RRset NSEC signé prouvant qu'il n'y a pas de RRset DS pour le nom délégué. Un résolveur sécurisé doit requêter les serveurs de nom pour la zone parent à la recherche du RRset DS si le référrant n'inclus ni un RRset DS ni un RRset NSEC prouvant que le RRset DS n'existe pas.

Si le validateur authentifie un RRset NSEC qui prouve qu'aucun RRset DS n'est présent pour cette zone, il n'y a pas de chemin d'authentification entre le parent et l'enfant. Si la résolveur a un DNSKEY initial ou un RR DS qui appartient à la zone enfant ou pour une

délégation sous la zone enfant, le RR initial peut être utilisé pour ré-établir un chemin d'authentification. Si un tel RR n'existe pas, le validateur ne peut pas authentifier les RRset.

Si le validateur ne support aucun algorithme listé dans le RRset DS authentifié, le résolveur n'a pas de chemin d'authentification. Le résolveur devrait traiter ce cas comme si un RRset NSEC authentifié prouvait qu'il n'y a pas de DS RRset.

Note que, pour une délégation signée, il y a 2 RR NSEC associés avec le nom délégué. Un RR NSEC réside dans la zone parent et peut être utilisé pour prouver si un RRset DS existe pour le nom délégué. Le second réside dans la zone enfant et identifie quels RRset sont présent à l'apex de la zone.

Si le résolveur ne supporte aucun algorithme listé dans un RRset DS, le résolveur n'est pas capable de vérifier le chemin d'authentification pour la zone enfant. Dans ce cas, le résolveur devrait traiter la zone enfant comme si elle était non-signée.

Authentifier un RRset avec un RR RRSIG

Un validateur peut utiliser un RR RRSIG et le RR DNSKEY correspondant pour tenter d'authentifier les RRset. Le validateur vérifie d'abord le RR RRSIG pour vérifier qu'il couvre le RRset, a un interval de temps valide, et identifier un RR DNSKEY valide. Le validateur construit ensuite la forme canonique de la donnée signée en ajoutant le RDATA RRSIG (sans le champ Signature) avec la forme canonique du RRset couvert. Finalement, le validateur utilise la clé publique et la signature pour authentifier la donnée signée.

Vérifier la validité RR RRSIG

Un résolveur sécurisé peut utiliser un RR RRSIG pour authentifier un RRset si toutes les conditions suivantes sont maintenues :

- Le RR RRSIG et le RRset doivent avoir le même nom propriétaire et la même classe
- Le champ Signer's Name du RR RRSIG doit être le nom de la zone qui contient le RRset
- Le champ Type Covered du RR RRSIG doit être égal au type du RRset.
- Le nombre de labels dans le nom propriétaire du RRset doit être supérieur ou égal à la valeur dans le champ Label du RR RRSIG
- La notion du validateur du temps courant doit être inférieur ou égal au temps listé dans le champ Expiration du RR RRSIG
- La notion du validateur du temps courant doit être supérieur ou égal au temps listé dans le champ Inception du RR RRSIG.
- Les champs Signer's Name, Algorithm, et Key Tag du RR RRSIG doivent correspondre au nom propriétaire, algorithm et key tag d'un RR DNSKEY dans le RRset DNSKEY de l'apex de zone.
- Le RR DNSKEY correspondant doit être présent dans le RRset DNSKEY de l'apex de zone, et doit avoir le bit Zone Flag mis.

Il est possible que plus d'un RR DNSKEY corresponde aux conditions ci-dessus. Dans ce cas, le validateur ne peut pas prédéterminer lequel utiliser pour authentifier la signature, et doit tenter chaque RR DNSKEY jusqu'à ce que la signature soit validée.

Noter que ce procéssus d'authentification est seulement significatif si le validateur authentifie le RR DNSKEY avant de l'utiliser pour valider les signatures. Le RR DNSKEY correspondant est considéré authentique si :

- Le RRset DNSKEY apex contenant le RR DNSKEY est considéré comme authentique, ou
- Le RRset couvert par le RR RRSIG est le RRset DNSKEY apex lui-même, et le RR DNSKEY match soit un RR DS authentifié de la zone parent ou matche une ancre de confiance.

Reconstruire les données signées

Une fois le RR RRSIG validé, le validateur doit reconstruire la donnée signée original. Cette donnée inclus le RDATA RRSIG (sans le champ signature) et la forme canonique du RRset. En plus d'être ordonné, la forme canonique peut également différer du RRset reçu à

cause de la compression de nom DNS, des TTLs décrémentés, ou de l'expansion wildcard. La validateur devrait utiliser le reconstruction suivante :

```
signed_data = RRSIG_RDATA | RR(1) | RR(2)...
```

où "|" dénote un concaténation.

```
RR(i) = name | type | class | OrigTTL | RDATA length | RDATA
```

name est calculé en accord avec la fonction plus bas

class est la classe du RRset

type est le type du RRset et de tous les RR dans la classe

OrigTTL est la valeur du champ Original TTL du RRSIG

- Tous les noms dans le champ RDATA sont sous la forme canonique
- Le jeu de RR(i) est trié par ordre canonique

Pour calculer le nom :

```
let rrsig_labels = La valeur du champ Labels RRSIG

let fqdn = nom de domaine pleinement qualifié sous la forme canonique

let fqdn_labels = compteur de label dans le fqdn

if rrsig_labels = fqdn_labels, name = fqdn

if rrsig_labels < fqdn_labels, name = "*." | Les labels rrsig_label les plus à droite du fqdn

if rrsig_labels > fqdn_labels
 le RR RRSIG ne passe pas nécessairement la validation et ne doit pas être utilisé pour authentifier ce
RRset.
```

Les RRset NSEC au point de délégation nécessitent un traitement spécial. Il y a 2 RRset NSEC distinct associé avec un nom signé délégué. Un RRset NSEC réside dans la zone parent, et spécifique quels RRset snt présent dans la zone parent. L'autre réside dans la zone enfant et identifie quels RRset sont présent à l'apex dans la zone enfant. En reconstruisant le RRset NSEC original puor la délégation depuis la zone parent, le RR NSEC ne doit pas être combiné avec les RR NSEC de la zone enfant. En reconstruisant le RRset NSEC original pour l'apex de la zone enfant, les RR NSEC ne doivent pas être combinés avec les RR NSEC de la zone parent.

Noter que les 2 RRset NSEC au point de délégation ont un RR RRSIG correspondant avec un nom propriétaire correspondant au nom délégué, et chacun de ces RR RRSIG est une donnée autoritative associée avec la zone qui contient le RRset NSEC.

Vérifier la signature

Une fois que le résolveur a validé le RR RRSIG et reconstruit les données signées originales, le validateur peut tenter d'utiliser la signature cryptographique pour authentifier les données signées, et donc finalement, authentifier le RRset.

Le champ Algorithm dans le RR RRSIG identifie l'algoritme cryptographique utilisé pour générer la signature. La signature elle-même est contenue dans le champ Signature du RDATA RRSIG, et la clé publique utilisée pour vérifier la signature est contenue dans le champ Public Key du ou des RR DNSKEY correspondants

Si le champ Labels du RR RRSIG n'est pas égal au nombre de labels dans le nom propriétaire pleinement qualifié du RRset, le RRest est soit invalide, ou le résultat d'une expansion wildcard. Le résolveur doit vérifier que l'expansion wildcard a été appliqué correctement avant de considérer le RRset comme authentique.

Si d'autre RR RRSIG couvrent également ce RRset, la stratégie de sécurité locale du résolveur détermine s'il doit tester les RR RRSIG et comment gérer les conflits si les RR RRSIG donnent différents résultats.

Si le résolveur accèpte le RRset, le validateur doit mettre le TTL du RR RRSIG et chaque RR dans le RRset authentifié à une valeur pas supérieur au minimum de :

- Le TTl du RRset reçu dans la réponse
- Le TTL du RR RRSIG reçu dans la réponse
- La valeur dans le champ Original TTL du RR RRSIG, et
- La différence du temps d'expiration de la signature du RR RRSIG et la date courante.

Authentifier un RRset wilcard étendu positif

Si le nombre de labels dans le nom propriétaire du RRset est supérieur au champ Labels du RR RRSIG couvrant, le RRset et sont RR RRSIG ont été créés en résultat d'une expansion wildcard. Une fois que le validateur a vérifié la signature, il doit vérifier la non-existance d'un match exacte pour la requête.

Noter que la réponse reçue par le résolveur devrait inclure tous les RR NSEC nécessaire pour authentifier la réponse.

Authentifier la non-existance

Un résolveur peut utiliser les RR NSEC authentifiés pour prouver qu'un RRset n'est pas présent dans une zone signée. Les serveurs de nom sécurisés devraient automatiquement inclure les RR NSEC nécessaires pour les zones signées dans leur réponses aux résolveurs sécurisés.

La non existance est déterminée par les règles suivantes :

- Si le RR demandé match le nom propriétaire d'un RR NSEC authentifié, le bit du type RR NSEC liste tous les types de RR présents au nom propriétaire, et un résolveur peut prouver que le type RR demandé n'existe pas en vérifiant le type RR dans le bit map. Si le nombre de labels dans un nom propriétaire du RR NSEC est égal au champ Labels du RR RRSIG couvrant, l'existence du RR NSEC prouve que l'expansion wildcard n'a pas été utilisé pour matcher la requête.
- Si le nom RR demandé apparaît après le nom propriétaire du RR NSEC authentifié et avant le nom listé dans le champ Next Domain du RR NSEC en accord avec l'ordre DNS canonique définis dans la rfc4034, alors aucun RRset avec le nom demandé n'existe dans la zone. Cependant, il est possible qu'un wildcard ait été utilisé pour correspondre au nom propriétaire et type du RR demandé, donc prouver que le RRset demandé n'existe pas nécessite également qu'aucun RRset wildcard possible n'existe qui pourrait être utilisé pour générer une résponse positive.

De plus, les résolveurs sécurisés doivent authentifier les RRset NSEC qui incluent la preuve de non-existence. Pour prouver la non-existence d'un RRset, le résolveur doit être capable de vérifier les RRset demandés qui n'existent pas et dont aucun RRset wildcard n'existe. Prouver cela peut nécessiter plus d'un RRset NSEC de la zone. Si le jeu complet de RRset NSEC n'est pas présent dans une réponse (par ex, dû à un message tronqué), le résolveur sécurisé doit renvoyer la demande pour pouvoir tenter d'obtenir la collection de RR NSEC nécessaire pour vérifier la non-existence de RRset demandé. Comme avec toutes les opération DNS, cependant, le résolveur doit délimiter le travail qu'il place dans la réponse d'une requête particulière.

Vu que le RR NSEC validé prouve la non-existence de lui-même et de ses RR RRSIG correspondants, un validateur doit ignorer les paramètres des bits NSEC et RRSIG dans un RR NSEC.

Comportement des résolveurs si les signatures échouent

Si aucun des RRSIG ne peut être validé, la résponse devrait être considérée BAD. Si la validation a été faite pour desservir une requête récursive, le serveur de nom doit retourner RCODE 2 au client. Cependant, il doit retourner la réponse complète si et seulement si la requête originale avait le bit CD mis.