Groupe de travail Réseau
Request for Comments : 2187
Catégorie : Information
Traduction Claude Brière de L'Isle

D. Wessels, K. Claffy
National Laboratory for Applied Network Research/UCSD
septembre 1997

Application du protocole des antémémoires Internet (ICP), version 2

Statut de ce mémoire
Le présent mémoire apporte des informations pour la communauté de l'Internet. Il ne spécifie aucune sorte de norme de l'Internet. La distribution du présent mémoire n'est soumise à aucune restriction.

Résumé
Le présent document décrit l'application du protocole des antémémoires de l'Internet (ICPv2, *Internet Cache Protocol version 2*) (RFC2186) à la mise en antémémoire sur la Toile. ICPv2 est un format de message léger utilisé pour les communications entre les antémémoires de la Toile. Plusieurs mises en œuvre indépendantes de mise en antémémoire utilisent maintenant ICP, et on considère qu'il est important de codifier les utilisations pratiques existantes de ICP pour ceux qui essayent de mettre en œuvre, déployer, et étendre son utilisation.

Les interrogations et réponses de ICP se réfèrent à l'existence des URL (ou objets) dans les antémémoires voisines. Les antémémoires échangent des messages ICP et utilisent les informations rassemblées pour choisir la localisation la plus appropriée d'où restituer un objet. Un document d'accompagnement (RFC2186) décrit le format et la syntaxe du protocole lui-même. Dans le présent document, nous nous concentrons sur les questions de déploiement d'ICP, de son efficacité, de sa sécurité, et de ses interactions avec les autres aspects du comportement du trafic de la Toile.

Table des matières

1. Introduction

ICP est un format de message léger utilisé pour communiquer entre les antémémoires de la Toile. ICP est utilisé pour échanger des indications sur l'existence des URL dans les antémémoires du voisinage. Les antémémoires échangent des interrogations et des réponses ICP pour rassembler des informations à utiliser dans le choix de la localisation la plus appropriée d'où restituer un objet.

Le présent document décrit la mise en œuvre de ICP dans le logiciel. Pour la description du protocole et du format de message, prière de se reporter au document d'accompagnement (RFC2186). Nous éviterons de porter des jugements sur la question de savoir si et comment ICP devrait être utilisé dans des configurations particulières d'antémémoires de la Toile. ICP peut être un "gain net" dans certaines situations et une "perte nette" dans d'autres. On reconnaît que certaines pratiques décrites dans ce document sont sous optimales. Certaines d'entre elles existent pour des raisons historiques. Certains aspects ont été améliorés dans les dernières versions. Comme le présent document sert seulement à décrire les pratiques actuelles, on se concentre plutôt sur la documentation que sur l'évaluation. Cependant, il s'adresse à des problèmes de sécurité reconnus et à d'autres insuffisances.

Le reste de ce document est rédigé comme suit. On commence par décrire la hiérarchie des antémémoires de la Toile, à expliquer les motifs de l'utilisation d'ICP, et à démontrer comment configurer son utilisation dans la hiérarchie des antémémoires. On donne ensuite une description pas à pas d'une transaction interrogation/réponse ICP. Puis on discute de l'interaction d'ICP avec les pare-feu, et une brève allusion à ICP en diffusion groupée. On termine par les leçons tirées durant le développement du protocole et son déploiement jusqu'à présent, puis les considérations canoniques sur la sécurité.

ICP a été initialement développé par Peter Danzig, et autres, à l'Université de Californie du Sud comme partie centrale d'une mise en mémoire hiérarchisée dans le projet de recherche Harvest [3].

2. Hiérarchies des antémémoires de la Toile

Une seule antémémoire de la Toile va réduire la quantité de trafic généré par les clients derrière elle. De même, un groupe d'antémémoires de la Toile peut largement tirer profit de la même façon du partage avec une autre antémémoire. Les chercheurs du projet Harvest envisageaient qu'il serait important de connecter hiérarchiquement les antémémoires de la Toile. Dans une hiérarchie (ou maillage) d'antémémoires, une antémémoire établit des relations d'homologue à homologue avec les antémémoires voisines. Il y a deux types de relations : parentes et sœurs. Une antémémoire parente est essentiellement un niveau au-dessus dans une hiérarchie d'antémémoires. Une antémémoire sœur est sur le même niveau. Les termes de "voisin" et "d'homologue" sont utilisés pour se référer soit aux parents soit aux sœurs qui sont à un seul "bond d'antémémoire" plus loin. La Figure 1 montre une configuration simple de hiérarchie.

Mais que veut dire être "sur le même niveau" ou "un niveau au dessus ?" Le flux général des demandes de document est au dessus de la hiérarchie. Lorsque une antémémoire ne détient pas un objet demandé, elle peut demander via ICP si une de ses antémémoires voisines a l'objet. Si une des voisines a bien l'objet demandé (c'est-à-dire, une "touche voisine") l'antémémoire va alors le lui demander. Si aucune des voisines n'a l'objet (un "échec de voisine") l'antémémoire doit transmettre la demande soit à une parente, soit directement au serveur d'origine. La différence essentielle entre une parente et une sœur est qu'une "touche de voisine" peut aller chercher dans l'une d'entre elles, mais un "échec de voisine" NE PEUT PAS aller chercher chez une sœur. En d'autre termes, dans une relation de sœur, une antémémoire peut seulement demander à restituer les objets que la sœur a déjà en antémémoire, tandis que la même antémémoire peut demander à une parente de restituer tout objet sans considérer si il est ou non mis en antémémoire. Le rôle d'une antémémoire parente est de fournir le "transit" pour la demande si nécessaire, et en conséquence, les antémémoires parentes sont idéalement situées au sein du ou sur le chemin d'un fournisseur d'accès Internet (FAI) de transit.

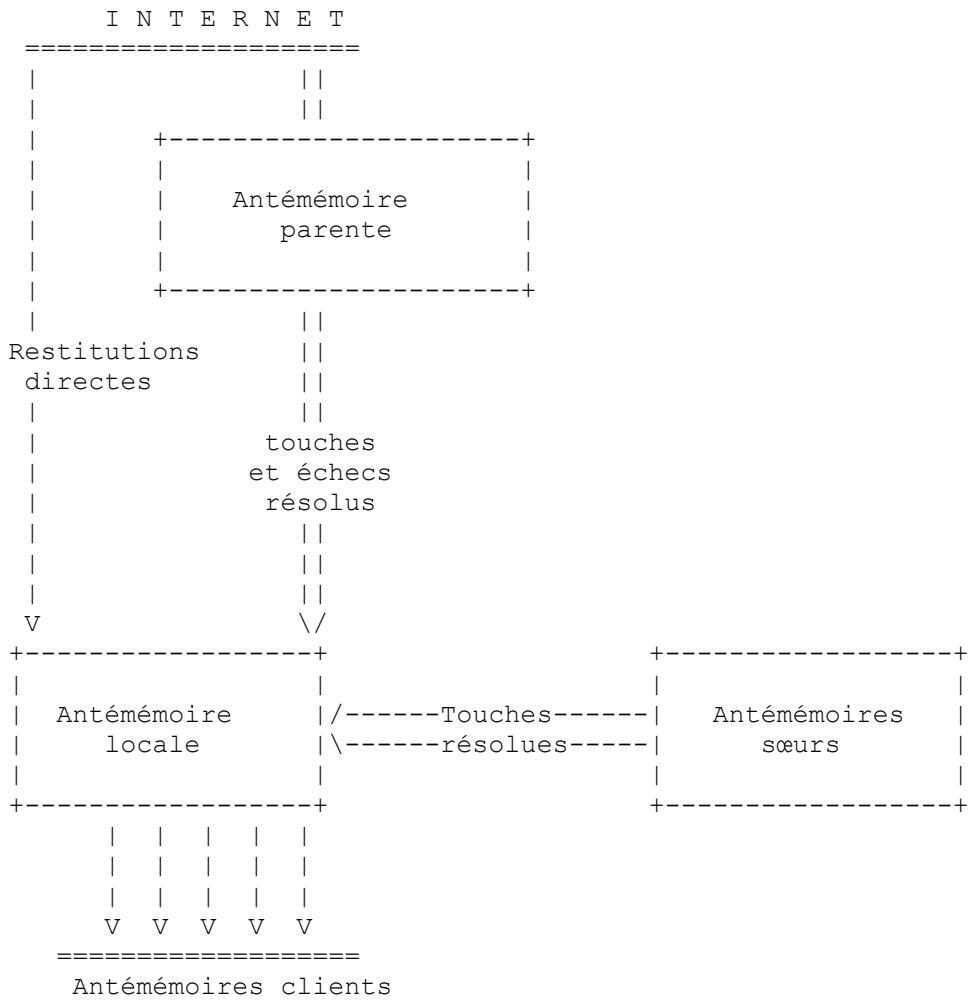

Figure 1 : Hiérarchie simple d'antémémoire de la Toile

L'antémémoire locale peut restituer des touches à partir des antémémoires sœurs, des touches et des échecs à partir des antémémoires parentes, et certaines demandes directement à partir des serveurs d'origine.

Squid et Harvest permettent des configurations hiérarchiques complexes. Par exemple, on peut spécifier qu'une voisine donnée sera utilisée seulement pour une certaine classe de demandes, telles que les URL d'un domaine spécifique du DNS.

De plus, il est possible de traiter une voisine comme une sœur pour certaines demandes et comme parente pour les autres.

Le modèle de hiérarchie d'antémémoires décrit ici comporte un certain nombre de dispositifs pour empêcher les antémémoires de niveau supérieur de devenir des points d'étranglement. L'un d'eux est la capacité à restreindre les parents à ce qui vient juste d'être décrit (par domaines). Une autre optimisation est que l'antémémoire ne transmet les demandes mettables en antémémoire qu'à ses voisines. Une large classe de demandes sur la Toile sont par nature non mettables en antémémoire ; parmi elles, les demandes qui exigent certains types d'authentification, les données chiffrées par la session, les réponses très personnalisées, et certains types d'interrogations de base de données. Les antémémoires de niveau inférieur devraient traiter directement ces demandes plutôt que de surcharger les antémémoires parentes.

3. Quelle est la valeur ajoutée de ICP ?

Bien qu'il soit possible d'entretenir des hiérarchies d'antémémoires sans utiliser ICP, l'absence de ICP ou de quelque chose de similaire interdit l'existence de relations de méta communication entre sœurs, c'est-à-dire de mécanismes pour interroger les antémémoires du voisinage sur un document donné.

Un souci de l'utilisation de ICP est le délai additionnel qu'introduit un échange d'interrogation/réponse ICP dans une transaction HTTP. Cependant, si l'interrogation ICP peut localiser l'objet dans une antémémoire voisine, le délai d'ICP peut être largement compensé par la plus rapide livraison des données à partir d'une voisine. Afin de minimiser les délais d'ICP, les antémémoires (aussi bien que le protocole lui-même) sont conçues pour retourner rapidement les demandes ICP. Bien sûr, l'application fait un traitement minimal de la demande ICP, la plus grande partie du délai relatif à ICP est dû à la transmission sur le réseau.

ICP sert aussi à fournir une indication de l'accessibilité des voisines. Si les réponses ICP provenant d'une voisine ne peuvent pas arriver, soit le chemin réseau est encombré (ou mort), soit l'application d'antémémoire ne fonctionne pas sur la machine voisine interrogée par ICP. Dans l'un et l'autre cas, l'antémémoire ne devrait pas utiliser cette voisine pour l'instant. De plus, comme une antémémoire inactive peut renvoyer les réponses plus rapidement que celle qui est occupée, tout cela s'équilibrant, ICP fournit une certaine forme d'équilibrage de charge.

4. Exemple de configuration de hiérarchie ICP

Configurer des antémémoires au sein d'une hiérarchie exige l'établissement de relations d'homologues à homologues, qui implique actuellement une configuration manuelle aux deux points d'extrémité homologues. Une antémémoire doit indiquer que l'autre est une parente ou une sœur. L'autre antémémoire va très vraisemblablement devoir ajouter la première antémémoire à ses listes de contrôle d'accès.

On montre ci-dessous quelques exemple de lignes de configuration dans une situation hypothétique. Nous avons deux antémémoires, une que fait fonctionner un FAI, et l'autre que fait fonctionner un consommateur. Nous décrivons d'abord comment le consommateur configurerait son antémémoire pour avoir des relations d'homologue avec le FAI. Puis nous décrivons comment le FAI permettrait au consommateur d'accéder à son antémémoire.

4.1 Configuration de l'antémémoire "proxy.customer.org"

Dans Squid, pour configurer parents et sœurs dans une hiérarchie, une directive `cache_host' est entrée dans le fichier de configuration. Le format est :

cache_host nom-d'hôte type accès-http accès-icp [options]

Où le type est "parent", "sœur" ou "diffusion groupée". Pour notre exemple, ce serait :

cache_host cache.fai.com parent 8080 3130

Cette configuration va amener l'antémémoire de l'utilisateur à résoudre la plupart des échecs d'antémémoire à travers le parent (les demandes "cgi-bin" et non GET seront résolues directement). Utiliser le parent peut être indésirable pour certains serveurs, comme les serveurs qui sont aussi dans le domaine customer.org. Pour toujours traiter de tels domaines locaux directement, le consommateur ajouterait ceci à son fichier de configuration :

local_domain customer.org

Il peut aussi être le cas que le consommateur veuille utiliser l'antémémoire du FAI seulement pour un sous ensemble spécifique de domaines du DNS. Le besoin de limiter les demandes de cette façon est en fait plus courant pour les niveaux supérieurs des hiérarchies d'antémémoire, mais il est néanmoins illustré ici. Pour limiter l'antémémoire du FAI à un sous ensemble de domaines du DNS, le consommateur utiliserait :

cache_host_domain cache.fai.com com net org

Alors, toutes les demandes qui NE SONT PAS dans les domaines .com, .net, ou .org seraient traitées directement.

4.2 Configuration de l'antémémoire "antémémoire.isp.com"

Pour configurer le côté qui reçoit les interrogations de la relation homologue d'antémémoires, on utilise des listes d'accès, similaires à celles utilisées dans les homologues d'acheminement. Les listes d'accès acceptent un large degré de personnalisation dans la relation d'homologues. Si aucune ligne d'accès n'est présente, l'antémémoire admet la demande par défaut.

Noter que l'antémémoire cache.isp.com n'a pas besoin de spécifier explicitement l'antémémoire du consommateur comme une homologue, ni le type de relation codé au sein de l'interrogation ICP elle-même. Les entrées de contrôle d'accès règlent la relation entre cette antémémoire et ses voisines. Pour notre exemple, le FAI utiliserait :

acl src Customer proxy.customer.org
http_access allow Customer
icp_access allow Customer

Cela définit une entrée de contrôle d'accès nommée "Customer" qui spécifie une adresse IP de source de la machine d'antémémoire du consommateur. L'antémémoire du consommateur serait alors admise à faire toute demande aux deux accès HTTP et ICP (y compris les échecs d'antémémoire). Cette configuration implique que l'antémémoire du FAI est une parente de celle du consommateur.

Si le FAI veut mettre en application une relation de sœur, il aura besoin de dénier l'accès aux échecs d'antémémoire. Cela serait fait comme suit :

miss_access deny Customer

Bien sûr, le FAI devrait aussi communiquer cela au consommateur, afin que le consommateur change sa configuration de parente à sœur. Autrement, si le consommateur demande un objet qui n'est pas dans l'antémémoire du FAI, un message d'erreur est généré.

5. Application du protocole

Les paragraphes qui suivent décrivent la mise en œuvre de ICP dans les paquetages d'antémémoires de la Toile de Harvest [3] (version de recherche) et de Squid [5]. En termes de numéro de version, cela signifie la version 1.4pl2 pour Harvest et la version 1.1.10 pour Squid.

La séquence de base des événements dans une transaction ICP est la suivante :
1. L'antémémoire locale reçoit une demande HTTP [1] de la part d'un client d'antémémoire.
2. L'antémémoire locale envoie des interrogations ICP (paragraphe 5.1).
3. La ou les antémémoires homologues reçoivent les interrogations et envoient les réponses ICP (paragraphe 5.2).
4. L'antémémoire locale reçoit les réponses ICP et décide où envoyer la demande (paragraphe 5.3).

5.1 Envoi des interrogations ICP

5.1.1 Déterminer si ICP doit être utilisé ou non

Toutes les demandes HTTP n'exigent pas l'envoi d'une interrogation ICP. Évidemment, les touches d'antémémoire n'auront pas besoin de ICP parce que la demande est satisfaite immédiatement. Pour les serveurs d'origine très proches de l'antémémoire, on ne veut pas utiliser les antémémoires voisines. Dans Squid et Harvest, l'administrateur spécifie ce qui constitue un serveur "local" avec les options de configuration "local_domain" et "local_ip". L'antémémoire contacte toujours un serveur local directement, et n'interroge jamais une antémémoire homologue.

Il y a d'autres classes de demandes que les antémémoires (ou leurs administrateurs) peuvent préférer transmettre directement au serveur d'origine. Dans Squid et Harvest, une de ces classes inclue toutes les méthodes de demande non GET. Une antémémoire Squid peut aussi être configurée pour ne pas utiliser les homologues pour les URL qui correspondent à la "hierarchy_stoplist".

Pour qu'une demande HTTP donne une transaction ICP, elle doit :
- ne pas être une touche d'antémémoire
- ne pas être pour un serveur local
- être une demande GET, et
- ne pas correspondre à la configuration "hierarchy_stoplist".

On appelle cela une demande "hiérarchique". Une demande "non hiérarchique" est celle qui ne génère aucun trafic ICP. Pour éviter de traiter des demandes qui diminueraient vraisemblablement l'efficacité de l'antémémoire, on peut configurer l'antémémoire pour ne pas consulter la hiérarchie pour les URL qui contiennent certaines chaînes (par exemple, "cgi_bin").

5.1.2 Déterminer quels homologues interroger

Par défaut, une antémémoire envoie un message ICP_OP_QUERY à chaque homologue, sauf si un de ce qui suit est vrai :

- Des restrictions empêchent d'interroger une homologue pour cette demande, sur la base de la directive de configuration "cache_host_domain", qui spécifie un ensemble de domaines DNS (d'après les URL) pour lesquels l'homologue devrait ou non être interrogée. Dans Squid, une directive plus souple ("antémémoire_host_acl") accepte des restrictions sur d'autres parties de la demande (méthode, numéro d'accès, source, etc.).

- L'homologue est une sœur, et la demande HTTP comporte un en-tête "Pragma: no-cache". Cela parce que la sœur se verrait demander de faire transiter la demande, ce qui n'est pas admis.

- L'homologue est configurée pour qu'on ne lui envoie jamais d'interrogations ICP (c'est-à-dire, avec l'option "no-query").

Si la détermination donne une seule homologue ICP interrogeable, et si la directive de configuration Squid "single_parent_bypass" est mise, on peut alors éviter d'attendre la seule réponse ICP et juste envoyer la demande HTTP directement à l'antémémoire homologue.

L'option de configuration Squid "source_ping" configure une antémémoire Squid pour envoyer un ping à la source originale simultanément à ses interrogations ICP, au cas où l'origine serait plus proche que toutes les antémémoires.

5.1.3 Calcul du nombre attendu de réponses ICP

Harvest et Squid veulent maximiser les chances d'obtenir une réponse HIT *(touche)* de l'une des homologues. Donc, l'antémémoire attend que toutes les réponses ICP soient reçues. Normalement, on s'attend à recevoir une réponse ICP pour chaque interrogation envoyée, sauf :

- Lorsque on pense que l'homologue est défaillante. Si l'homologue est défaillante, Squid et Harvest continuent de lui envoyer des interrogations ICP, mais n'attendent pas qu'elle réponde. Lorsque une réponse ICP est de nouveau reçue de cette homologue, son état sera changé en actif.

  La détermination de l'état actif/inactif a varié un petit peu avec l'évolution du logiciel Harvest et Squid. Harvest et Squid marquent tous deux une homologue comme inactive lorsque elle échoue à répondre à 20 interrogations ICP consécutives. Squid marque aussi une homologue comme inactive lorsque une connexion TCP échoue, et active à nouveau lorsque un diagnostic de connexion TCP réussit.

- Lors d'un envoi à une adresse de diffusion groupée. Dans ce cas, on va probablement s'attendre à recevoir plus d'une réponse, et on n'a aucun moyen de déterminer à combien s'attendre. On discute des questions de diffusion groupée à la section 7 ci-dessous.

5.1.4 Installation d'un événement de fin de temporisation

Parce que ICP utilise UDP comme transport sous-jacent, les interrogations et les réponses ICP peuvent parfois être abandonnées par le réseau. L'antémémoire installe un événement de fin de temporisation dans le cas où toutes les réponses attendues n'arriveraient pas. Par défaut, Squid et Harvest utilisent une temporisation de deux secondes. Si la restitution de l'objet n'a pas commencé lorsque survient la fin de temporisation, une source est choisie comme décrit au paragraphe 5.3.9 ci-dessous.

5.2 Réception des interrogations ICP et envoi des réponses

Lorsque un message ICP_OP_QUERY est reçu, l'antémémoire l'examine et décide quel message de réponse doit être envoyé. Elle va envoyer un des opcodes de réponse suivants, essayés dans l'ordre de la liste.

5.2.1 ICP_OP_ERR

L'URL est extrait de la charge utile et analysé. Si l'analyse échoue, un message ICP_OP_ERR est retourné.

5.2.2 ICP_OP_DENIED

Les contrôles d'accès sont vérifiés. Si l'homologue n'est pas admise à faire cette demande, ICP_OP_DENIED est retourné. Squid compte le nombre de messages ICP_OP_DENIED envoyés pour chaque homologue. Si plus de 95 % sur plus de 100 réponses ont été rejetées, aucune réponse n'est envoyée. Cela empêche les antémémoires mal configurées d'envoyer sans fin des messages ICP inutiles.

5.2.3 ICP_OP_HIT

Si l'antémémoire atteint ce point sans encore correspondre à un des opcodes précédents, cela signifie que la demande est admise et on doit déterminer si ce sera un HIT *(touche)* ou un MISS *(échec)*, et on vérifie donc si l'URL existe dans l'antémémoire locale. Si elle existe et si l'entrée d'antémémoire est fraîche pour au moins les prochaines 30 secondes, on peut retourner un message ICP_OP_HIT. La détermination de périmée/fraîche utilise les règles de rafraîchissement (ou de TTL) locales.

Noter qu'il existe une condition de compétition pour les réponses ICP_OP_HIT aux homologues sœurs. La réponse ICP_OP_HIT signifie qu'une demande HTTP suivante pour l'URL désigné résulterait en une touche d'antémémoire. On suppose que la demande HTTP va venir très vite après la ICP_OP_HIT. Cependant, il existe une petite probabilité que l'objet soit purgé de cette antémémoire avant la réception de la demande HTTP. Si cela survient, et si l'homologue qui répond a appliqué la configuration "miss_access" de Squid, l'usager va recevoir un message de refus d'accès très embarrassant.

5.2.3.1 ICP_OP_HIT_OBJ

Avant de retourner le message ICP_OP_HIT, on regarde si on peut envoyer à la place un message ICP_OP_HIT_OBJ. On peut utiliser ICP_OP_HIT_OBJ si :

- o le message ICP_OP_QUERY avait le fanion ICP_FLAG_HIT_OBJ mis,
- o l'objet tout entier (plus l'URL) va tenir dans un message ICP. La taille maximum de message ICP est de 16 k octet, mais une application peut choisir de régler à un plus faible maximum les réponses ICP_OP_HIT_OBJ.

Normalement les réponses ICP sont envoyées immédiatement après la réception de l'interrogation, mais le message ICP_OP_HIT_OBJ ne peut pas être envoyé tant que les données d'objet ne sont pas disponibles pour être copiées dans le message de réponse. Pour Squid et Harvest cela signifie que l'objet doit être "transféré" du disque si il n'est pas déjà en mémoire. Donc, en moyenne, une réponse ICP_OP_HIT_OBJ aura une plus forte latence qu'une ICP_OP_HIT.

5.2.4 ICP_OP_MISS_NOFETCH

À ce point nous avons un échec d'antémémoire. ICP a deux types de réponses d'échec. Si l'antémémoire ne veut pas que l'homologue lui demande l'objet, elle envoie un message ICP_OP_MISS_NOFETCH.

5.2.5 ICP_OP_MISS

Finalement, une réponse ICP_OP_MISS est retournée par défaut. Si l'antémémoire qui répond est une parente de l'antémémoire qui interroge, la réponse ICP_OP_MISS indique une invitation à aller chercher l'URL à travers l'antémémoire qui répond.

5.3 Réception des réponses ICP

Certaines réponses ICP seront ignorées ; en particulier, lorsque une des conditions suivantes est vraie :

- o Le message de réponse provient d'un homologue inconnu.
- o L'objet désigné par l'URL n'existe pas.
- o On est déjà allé chercher l'objet.

5.3.1 ICP_OP_DENIED

Si plus de 95 % de plus de 100 réponses provenant d'une antémémoire homologue ont eu ICP_OP_DENIED, un tel taux de refus indique très vraisemblablement une erreur de configuration, soit locale soit chez l'homologue. Pour cette raison, aucune autre interrogation ne sera envoyée à l'homologue pour la durée du fonctionnement de l'antémémoire.

5.3.2 ICP_OP_HIT

La restitution d'objet commence immédiatement à partir de l'homologue qui répond.

5.3.3 ICP_OP_HIT_OBJ

Les données d'objet sont extraites du message ICP est la restitution est achevée. Si il y a des problèmes avec le message ICP_OP_HIT_OBJ (par exemple des données manquantes) la réponse sera traitée comme une ICP_OP_HIT standard.

5.3.4 ICP_OP_SECHO

La restitution d'objet commence immédiatement à partir du serveur d'origine parce que la réponse ICP_OP_SECHO est arrivée avant toute ICP_OP_HIT. Si une ICP_OP_HIT était arrivée avant, cette réponse ICP_OP_SECHO aurait été ignorée parce que la restitution aurait déjà commencé.

5.3.5 ICP_OP_DECHO

Une réponse ICP_OP_DECHO est traités comme une ICP_OP_MISS. Les homologues non ICP doivent toujours être configurées comme des parentes ; une sœur non ICP n'a pas de sens. Un sérieux problème avec le dispositif ICP_OP_DECHO est que comme il lance des messages à partir de l'accès d'écho UDP de l'homologue, il n'indique pas que

l'antémémoire de l'homologue est en cours de fonctionnement – mais seulement que la connexité réseau existe entre les homologues.

5.3.6 ICP_OP_MISS

Si l'homologue est une sœur, la réponse ICP_OP_MISS est ignorée. Autrement, l'homologue peut être "remémorée" pour utilisation future au cas où aucune réponse de touche *(HIT)* ne serait reçue ensuite(paragraphe 5.3.9).

Harvest et Squid rappellent au premier parent de retourner un message ICP_OP_MISS. Avec Squid, les parents peuvent être pondérés de telle sorte que les "premiers parents à échouer" peuvent ne pas être réellement la première réponse reçue. Nous appelons cela FIRST_PARENT_MISS. On se souvient que les échec de sœur sont entièrement ignorés ; on ne se soucie que des échecs des parents. Les RTT d'échec des parent peuvent être pondérés parce que quelque fois le plus proche parent n'est pas celui que les gens veulent utiliser.

Aussi, les versions récentes de Squid peuvent rappeler au parent le plus faible RTT vers le serveur d'origine, en utilisant l'option ICP_FLAG_SRC_RTT. On appelle cela le CLOSEST_PARENT_MISS.

5.3.7 ICP_OP_MISS_NOFETCH

Cette réponse est essentiellement ignorée. Une antémémoire ne doit pas transmettre une demande à une homologue qui retourne ICP_OP_MISS_NOFETCH.

5.3.8 ICP_OP_ERR

Ignorée en silence.

5.3.9 Lorsque toutes les homologues échouent

Pour ICP_OP_HIT et ICP_OP_SECHO, la demande est immédiatement transmise. Pour ICP_OP_HIT_OBJ il n'est pas besoin de transmettre la demande. Pour tous les autres opcodes de réponse, on attend jusqu'à ce que le nombre de réponses attendues ait été reçu. Lorsque on a toutes les réponses attendues, ou lorsque survient la fin de temporisation de l'interrogation, il est temps de transmettre la demande.

Comme toutes les réponses MISS *(échec)* ont été reçues de toutes les homologues, on doit choisir soit une antémémoire parente soit le serveur d'origine.

- Si les homologues utilisent le dispositif ICP_FLAG_SRC_RTT, on transmet la demande à l'homologue qui a le plus faible RTT vers le serveur d'origine. Si l'antémémoire locale mesure aussi les RTT vers les serveurs d'origine, et qu'elle est plus proche qu'aucune des parentes, la demande est transmise directement au serveur d'origine.

- Si il y a un parent FIRST_PARENT_MISS disponible, la demande lui sera transmise.

- Si l'échange d'interrogation/réponse ICP n'a pas produit de parente appropriée, la demande sera envoyée directement au serveur d'origine (sauf si des restrictions liées à des pare-feu l'empêchent).

5.4 Options ICP

Les options suivantes ont été ajoutés à Squid pour prendre en charge de nouvelles caractéristiques tout en maintenant la rétro compatibilité avec la mise en œuvre Harvest.

5.4.1 ICP_FLAG_HIT_OBJ

Ce fanion est à zéro par défaut et ne sera établi dans un message ICP_OP_QUERY que si les trois critères suivants sont satisfaits :

- Il est activé dans le fichier de configuration d'antémémoire avec "udp_hit_obj on".
- L'homologue doit utiliser ICP version 2.
- La demande HTTP ne doit pas inclure l'en-tête "Pragma: no-cache".

5.4.2 ICP_FLAG_SRC_RTT

Ce fanion est à zéro par défaut et ne sera établi dans un message ICP_OP_QUERY que si ces deux critères sont satisfaits :

- Il est activé dans le fichier de configuration d'antémémoire avec "query_icmp on".
- L'homologue doit utiliser ICP version 2.

6. Pare-feu

Faire fonctionner une antémémoire de la Toile derrière un pare-feu ou dans un réseau privé pose quelques problèmes intéressants. La partie difficile est d'arriver à trouver si l'antémémoire est capable de se connecter au serveur d'origine. Harvest et Squid fournissent une directive de configuration "inside_firewall" pour faire la liste des domaines du DNS sur le côté proche d'un pare-feu. Tout le reste est supposé être sur le côté distant d'un pare-feu. Squid a aussi une directive "firewall_ip" qui fait que les hôtes intérieurs peuvent aussi être spécifiés par les adresses IP.

Dans une configuration simple, une antémémoire Squid derrière un pare-feu aura seulement une antémémoire parente (qui est sur le pare-feu lui-même). Dans ce cas, Squid doit utiliser cette parente pour tous les serveurs au delà du pare-feu, de sorte qu'il n'est nul besoin d'utiliser ICP.

Dans une configuration plus complexe, il peut y avoir un certain nombre d'antémémoires homologues qui sont aussi derrière le pare-feu. Ici, ICP peut être utilisé pour vérifier les touches d'antémémoire dans les homologues. À l'occasion, lorsque ICP est utilisé, il se peut qu'aucune réponse ne soit reçue. Si l'antémémoire n'avait pas été derrière un pare-feu, la demande aurait été transmise directement au serveur d'origine. Mais dans notre situation, l'antémémoire doit trouver une antémémoire parente, soit de façon aléatoire soit par des informations de configuration. Par exemple, Squid admet qu'une antémémoire parente soit désignée comme choix par défaut lorsque aucune autre n'est disponible.

7. Diffusion groupée

Pour une distribution efficace, une antémémoire peut livrer les interrogations ICP à une adresse de diffusion groupée, et les antémémoires voisines peuvent se joindre au groupe de diffusion groupée pour recevoir de telles interrogations.

La pratique actuelle est que les antémémoires n'envoient les réponses ICP qu'aux adresses en envoi individuel, pour plusieurs raisons :

- Envoyer les réponses ICP en diffusion groupée ne réduirait pas le nombre de paquets envoyés.
- Cela empêche les autres membres du groupe de recevoir des réponses inattendues.
- La réponse devrait suivre les chemins d'acheminement d'envoi individuel pour indiquer la connexité (en envoi individuel) entre le receveur et l'envoyeur car la demande HTTP qui va suivre sera acheminée en envoi individuel.

La confiance est un aspect important des relations inter antémémoires. Une antémémoire de la Toile ne devrait pas automatiquement faire confiance à toute antémémoire qui répond à une interrogation ICP en diffusion groupée. Les antémémoires devraient ignorer les messages ICP provenant d'adresses non spécifiquement configurées comme voisines. Autrement, on pourrait facilement polluer un maillage d'antémémoires en faisant fonctionner une antémémoire illégitime et en lui faisant rejoindre un groupe, retourner ICP_OP_HIT pour toutes les demandes, et livrer ensuite un contenu frelaté.

Lors des envois à des groupes de diffusion groupée, les administrateurs d'antémémoire doivent faire attention à utiliser le TTL de diffusion groupée minimum requis pour atteindre tous les membres du groupe. Aucun privilège particulier n'est exigé pour se joindre à un groupe de diffusion groupée et il n'y a aucun moyen pour empêcher quiconque de se joindre à "votre" groupe. Deux groupes d'antémémoires utilisant la même adresse de diffusion groupée pourraient se chevaucher, ce qui serait cause qu'une antémémoire recevrait des réponses ICP de voisines inconnues. Les voisines inconnues ne seraient pas utilisées pour restituer les données d'objet, mais l'antémémoire recevrait constamment des réponses ICP qu'elle devrait toujours ignorer.

Pour empêcher un chevauchement de maillage d'antémémoires, les antémémoires devraient donc limiter la portée de leurs interrogations ICP avec les TTL approprié ; une application telle que mtrace[6] peut déterminer les TTL de diffusion groupée appropriés.

Comme mentionné au paragraphe 5.1.3, on a besoin d'estimer le nombre de réponses attendues pour un message ICP_OP_QUERY. Pour un envoi individuel, on attend une seule réponse pour chaque interrogation si l'homologue est active. Cependant, pour la diffusion groupée on attend généralement plus d'une réponse, mais on n'a aucun moyen de savoir exactement combien de réponses attendre. Squid envoie régulièrement (toutes les 15 minutes) des messages d'essai ICP_OP_QUERY aux seules homologues de groupes de diffusion groupée. Comme avec une interrogation ICP réelle, un événement de fin de temporisation est installé et les réponses sont comptées jusqu'à ce que la fin de temporisation survienne. Nous avons trouvé que le compte reçu varie considérablement. Donc, le nombre de réponses à attendre est calculé comme une moyenne mobile, arrondie à l'entier inférieur le plus proche.

8. Leçons tirées

8.1 Différences entre ICP et HTTP

ICP présente des différences notables avec HTTP. HTTP prend en charge un ensemble riche et sophistiqué de dispositifs. À l'inverse, ICP a été conçu pour être simple, petit, et efficace. Les en-têtes de demande et réponse HTTP consistent en lignes de texte ASCII délimité par une paire de CRLF, tandis que ICP utilise un en-tête de taille fixe et représente les nombres en binaire. La seule chose qu'ICP et HTTP ont en commun est l'URL.

Noter que le message ICP ne comporte même pas la méthode de demande de HTTP. La mise en œuvre d'origine supposait que seules les demandes GET seraient mettables en antémémoire et qu'il ne serait pas besoin de localiser des demandes non GET dans des antémémoires voisines. Et donc, la version actuelle de ICP ne s'accommode pas des demandes non GET, bien que la prochaine version de ce protocole inclura vraisemblablement un champ pour la méthode de demande.

HTTP définit des dispositifs qui sont importants pour la mise en antémémoire mais ne sont pas exprimables avec le protocole ICP actuel. Parmi eux sont "Pragma: no-cache", "If-Modified-Since", et tous les dispositifs de "Cache-Control" de HTTP/1.1. Un message ICP_OP_HIT_OBJ peut livrer un objet qui peut ne pas obéir aux contraintes de l'en-tête de demande. Ces différences entre ICP et HTTP sont la raison pour laquelle nous déconseillons l'utilisation du dispositif ICP_OP_HIT_OBJ.

8.2 Parents, sœurs, touches et échecs

Noter que les messages ICP n'ont pas de champ pour indiquer les intentions de l'antémémoire qui interroge. C'est-à-dire que nulle part dans la demande ou la réponse ICP il n'est dit que les deux antémémoires ont une relation de sœur ou de parent. Une antémémoire sœur peut seulement répondre par un HIT ou MISS, et non avec "tu peux restituer ceci à partir de chez moi" ou "tu ne peux pas récupérer ceci chez moi". L'antémémoire qui interroge doit appliquer la réponse HIT ou MISS à sa configuration locale pour l'empêcher de résoudre les échecs à travers une antémémoire sœur. Cette contrainte est étrange, parce que cet aspect de la relation ne peut être configuré que dans l'antémémoire d'origine des demandes, et indirectement via les contrôles d'accès configurés dans l'antémémoire interrogée comme décrit plus haut au paragraphe 4.2.

8.3 Différents rôles de ICP

On peut comprendre de deux façons différentes ce qu'est exactement le rôle de ICP dans un maillage d'antémémoires. Une acception est que le rôle de ICP est seulement la localisation des objets, précisément, de fournir des indications sur l'existence ou non de l'objet désigné dans une antémémoire voisine. Une hypothèse implicite est que les touches d'antémémoire sont très désirables, et ICP est utilisé pour maximiser les chances de les obtenir. Si un message ICP est perdu à cause de l'encombrement, rien de significatif n'est perdu ; la demande sera satisfaite de toute façon.

Il est de plus en plus demandé à ICP de remplir un rôle plus complexe : convoyer les politiques d'usage des antémémoire. Par exemple, de nombreuses organisations (par exemple, des universités) vont installer une antémémoire de la Toile sur la frontière de leur réseau. De telles organisations peuvent être heureuses d'établir des relations de sœur avec d'autres antémémoires du voisinage, sous réserve des conditions suivantes :

- o tout consommateur ou usager de l'organisation peut demander tout objet (en antémémoire ou non),
- o tout le monde peut demander un objet qui est déjà dans l'antémémoire,
- o tout le monde peut demander tout objet aux serveurs de l'organisation derrière l'antémémoire,.
- o toutes les autres demandes sont refusées ; en particulier, l'organisation ne fournit pas le transit aux demandes dans lesquelles ni le client ni le serveur ne tombent dans son domaine.

Pour convoyer avec succès les politiques, l'échange ICP doit prédire très précisément le résultat (touche, échec) d'une demande HTTP ultérieure. Le résultat peut souvent dépendre d'autres champs de demande, tels que Cache-Control. Il n'est donc pas possible à ICP de prédire précisément le résultat sans plus de la demande HTTP, sinon tout.

8.4 Fautes de conception du protocole ICPv2

On reconnaît certaines fautes de la conception originale de ICP, et on les note afin que les versions futures puissent éviter les mêmes erreurs.

- o L'URL terminé par NULL dans le champ de charge utile exige de progresser à travers le message un octet à la fois pour trouver certains des champs (c'est-à-dire, le début des données d'objet dans un message ICP_OP_HIT_OBJ).

- o Deux champs (Adresse de l'hôte envoyeur et Adresse de l'hôte demandeur) sont spécifiques de IPv4. Cependant, aucun

de ces champs n'est utilisé en pratique ; ils sont normalement remplis de zéros. Si les adresses IP ont un rôle dans le message ICP, il y a besoin d'un descripteur de famille d'adresse pour chacune d'elles, et les clients doivent être capables de dire si ils veulent ou non entendre les réponses IPv6.

- Les options sont limitées à 32 fanions d'option et à 32 bits de données d'option. Cela devrait être assez semblable à TCP, avec un descripteur d'option suivi par les données d'option.

- Bien qu'actuellement utilisée comme la clé de l'antémémoire, la chaîne d'URL ne tient plus adéquatement ce rôle. Certaines réponses HTTP varient maintenant selon l'agent d'utilisateur et autres en-têtes du demandeur. Une clé d'antémémoire doit incorporer tous les en-têtes non transport présents dans la demande du client. Tous les en-têtes non bond par bond de la demande devraient être envoyés dans une interrogation ICP.

- ICPv2 utilise des valeurs d'opcodes différentes pour les interrogations et les réponses. ICP devrait utiliser le même opcode pour les deux côtés d'une transaction entre deux, avec un indicateur "interrogation/réponse" disant avec quel côté elle est.

- ICPv2 ne comporte pas de champs d'authentification.

9. Considérations pour la sécurité

La sécurité pose problème avec ICP sur UDP à cause de sa nature sans connexion. Nous examinons ci-dessous diverses faiblesses et méthodes d'attaque, ainsi que leurs implications.

Notre première ligne de défense est de vérifier l'adresse IP de source du message ICP, par exemple, telle que donnée par recvfrom(2). Les messages d'interrogation ICP devraient être traités si les règles de contrôle d'accès permettent à l'adresse de celui qui interroge d'accéder à l'antémémoire. Cependant, les messages de réponse ICP doivent seulement être acceptés s'ils proviennent de voisines connues ; une antémémoire doit ignorer les réponses provenant d'adresses inconnues.

Parce qu'on fait confiance à la validité d'une adresse dans un paquet IP, ICP est susceptible d'usurpation d'adresse IP. Dans ce document, nous exposons certaines des conséquences de l'usurpation d'adresse IP. Normalement, les adresses usurpées ne peuvent être détectée que par les routeurs, et pas par les hôtes. Cependant, l'en-tête IP Authentication [7, 8] peut être utilisé en dessous d'ICP pour fournir une authentification cryptographique de la totalité du paquet IP qui contient le protocole ICP, éliminant ainsi le risque d'usurpation d'adresse IP.

9.1 Insertion d'interrogations ICP fautives

Le traitement d'un message ICP_OP_QUERY n'a pas d'implication connue pour la sécurité, pour autant que l'adresse demandeuse reçoive l'accès à l'antémémoire.

9.2 Insertion de réponses ICP fautives

Nous avons ici le problème d'un tiers qui génère des messages de réponse ICP qui sont retournés à l'antémémoire qui interroge avant que la réponse réelle n'arrive, ou avant qu'aucune réponse n'arrive. Le tiers peut insérer des réponses ICP fautives qui paraissent provenir de voisines légitimes. Il y a trois faiblesses :

- Empêcher une certaine voisine d'être utilisée
  Si un tiers pouvait renvoyer une réponse ICP_OP_MISS_NOFETCH avant l'arrivée de la réponse réelle, la voisine (falsifiée) ne serait pas utilisée.

  Un tiers pourrait inonder une antémémoire avec des messages ICP_OP_DENIED jusqu'à ce que soit atteint le seuil décrit au paragraphe 5.3.1, causant par là l'interruption temporaire de la relation de voisinage.

- Forcer à l'utilisation d'une certaine voisine
  Si un tiers pouvait renvoyer une réponse ICP_OP_HIT avant l'arrivée de la réponse réelle, la voisine (falsifiée) serait utilisée. Cela peut violer les termes d'une relation de sœur ; les réponses ICP_OP_HIT signifient qu'une demande HTTP ultérieure sera aussi une touche.

  De même, si des messages ICP_OP_SECHO falsifiés peuvent être générés, l'antémémoire va restituer les demandes directement du serveur d'origine.

- Empoisonnement d'antémémoire
  Le message ICP_OP_HIT_OBJ est particulièrement sensible aux questions de sécurité car il contient les données

d'objet réelles. En combinaison avec l'usurpation d'adresse IP, cette option ouvre la possibilité que l'antémémoire soit polluée par des objets invalides.

9.3 Espionnage

Les interrogations ICP en diffusion groupée donnent une méthode très simple pour "espionner" les messages ICP. Si ils activent la diffusion groupés, les administrateurs d'antémémoire devraient configurer l'application à utiliser le TTL minimum de diffusion groupée requis, en utilisant un outil tel que mtrace[6]. Noter que le mécanisme d'encapsulation dans IP de charge utile de sécurité (ESP) [7,9] peut être utilisé pour protéger contre l'espionnage des messages ICP.

L'espionnage du trafic ICP peut fournir à des tiers une liste des URL qui sont recherchés par les utilisateurs d'antémémoire. Comme l'adresse de l'hôte demandeur est remplie de zéros par Squid et Harvest, les URL ne peuvent pas être retransposés en systèmes d'hôtes individuels.

Par défaut, Squid et Harvest n'envoient pas de messages ICP pour les URL qui contiennent "cgi-bin" ou "?". Ces URL contiennent parfois des informations sensibles comme paramètres d'argument. Les administrateurs d'antémémoire doivent savoir que l'altération de la configuration pour faire des interrogations ICP pour de tels URL peut exposer des informations sensibles à des étrangers, en particulier lorsque la diffusion groupée est utilisée.

9.4 Blocage des messages ICP

Les interrogations ou réponses ICP intentionnellement bloquées (ou éliminées) vont apparaître pour refléter des défaillances ou encombrement de liaison, et vont empêcher l'utilisation d'une voisine aussi bien que conduire à des expirations de temporisations (voir au paragraphe 5.1.4). Si tous les messages sont bloqués, l'antémémoire va supposer que la voisine est défaillante et la retirer de son algorithme de sélection. Cependant, si, par exemple, toutes les autres interrogations sont bloquées, la voisine va rester "active," mais toutes les autres demandes vont subir la fin de temporisation ICP.

9.5 Retarder les messages ICP

L'algorithme de sélection de voisine attend normalement que toutes les réponses MISS ICP arrivent. Retarder les interrogations ou les réponses, afin qu'elles arrivent plus tard qu'elles ne l'auraient fait normalement va causer des retards supplémentaires pour la demande HTTP ultérieure. Bien sûr, si les messages sont tellement retardés qu'ils arrivent après la fin de temporisation, le comportement est le même que celui du "blocage" ci-dessus.

9.6 Déni de service

Une attaque de déni de service, où l'accès ICP est inondé d'un flux continuel de messages fautifs présente trois faiblesses :
- o l'application peut enregistrer tous les messages ICP fautifs et finalement provoquer une partition de disque ;
- o la file d'attente de prise de réception peut se remplir, causant l'abandon de messages légitimes ;
- o l'hôte peut gaspiller plusieurs cycles de CPU à recevoir les messages fautifs.

9.7 Altération des champs ICP

Nous supposons ici qu'un tiers est capable de changer un ou plusieurs des champs du message de réponse ICP.

Opcode
Changer le champ opcode est assez semblable à insérer des messages falsifiés décrit ci-dessus. Changer une touche en échec empêcherait l'homologue d'être utilisée. Changer un échec en touche forcerait l'utilisation de l'homologue.

Version
Altérer le champ Version ICP peut avoir des conséquences imprévisibles si le nouveau numéro de version est reconnu et pris en charge. L'application receveuse devrait ignorer les messages qui ont des numéros de version invalides. Au moment de cette rédaction, les numéros de version 2 et 3 sont tous deux utilisés. Ces deux versions utilisent certains champs (par exemple, Options) d'une manière légèrement différente.

Longueur de message
Une longueur de message incorrecte devrait être détectée par l'application receveuse comme message ICP invalide.

Numéro de demande
Le numéro de demande est souvent utilisé au titre de la clé d'antémémoire. Harvest n'utilise pas le numéro de demande. Squid utilise le numéro de demande en conjonction avec l'URL pour créer la clé d'une antémémoire. Altérer le numéro de demande va causer un échec de la recherche de la clé d'antémémoire. Ceci est similaire au blocage de réponse ICP.

Il n'est pas exigé qu'une antémémoire utilise à la fois l'URL et le numéro de demande pour localiser les demandes HTTP avec les interrogations ICP en instance (cependant, Squid et Harvest le font tous deux). Le numéro de demande doit toujours être le même dans l'interrogation et la réponse. Cependant, si l'antémémoire qui interroge utilise seulement le numéro de demande pour localiser les demandes en cours, il y a une possibilité que l'antémémoire qui répond incrémente le numéro de demande dans la réponse pour donner la fausse impression que les deux antémémoires sont plus proches qu'elles ne le sont en réalité. En d'autres termes, en supposant qu'il y a toujours quelques demandes ICP "en l'air" à tout moment, incrémenter le numéro de demande de la réponse trompe l'antémémoire qui interroge en lui faisant croire à un plus petit délai d'aller-retour que ce qui existe réellement.

Options
Il y a peu de risque de voir les champs binaires Options altérés. Un bit d'option ne doit être établi dans une réponse que si il était déjà établi dans l'interrogation. Changer un bit de zéro à un est détectable par l'antémémoire qui interroge, et un tel message doit être ignoré. Changer un bit de un à zéro est permis et n'a pas d'effet secondaire négatif.

Données d'option
ICP_FLAG_SRC_RTT est la seule option qui utilise le champ Données d'option. Altérer les valeurs de RTT retournées ici peut affecter l'algorithme de sélection de voisine, soit en forçant soit en empêchant l'utilisation d'une voisine.

URL
L'URL et le numéro de demande sont utilisés pour générer la clé d'antémémoire. Altérer l'URL va cause un échec de la recherche de la clé de l'antémémoire, et la réponse ICP sera entièrement ignorée. Ceci est similaire au blocage de la réponse ICP.

9.8 Résumé

- o ICP_OP_HIT_OBJ est particulièrement vulnérable aux problèmes de sécurité parce qu'il comporte des données d'objet. Pour cette raison, et d'autres, son utilisation est déconseillée.

- o Falsifier, altérer, insérer, ou bloquer des messages ICP peut cause l'échec d'une demande HTTP dans seulement deux situations :
- si l'antémémoire est derrière un pare-feu et ne peut pas se connecter directement au serveur d'origine ;
- si une fausse réponse ICP_OP_HIT cause la transmission de la demande HTTP à une sœur lorsque la demande est un échec d'antémémoire et que la sœur refuse de continuer à transmettre la demande au nom de l'antémémoire d'origine.

- o Falsifier, altérer, insérer, ou bloquer des messages ICP peut facilement causer la transmission (ou la non transmission) des demandes HTTP à certaines voisines. Si l'antémémoire voisine a aussi été compromise, elle pourrait alors servir des contenus falsifiés et polluer une hiérarchie d'antémémoires.

- o Bloquer ou retarder des messages ICP peut causer un retard encore plus grand de la demande HTTP, mais pas l'arrêter.

10. Références

[1] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, T. Berners-Lee, "Protocole de transfert Hypertext -- HTTP/1.1", RFC2068, janvier 1997. (*Obsolète, voir* RFC2616) *(P.S.)*

[2] T. Berners-Lee, L. Masinter et M. McCahill, "Localisateurs uniformes de ressource (URL)", décembre 1994. *(Obsolète, voir les RFC 4248 et 4266)*

[3] Bowman M., Danzig P., Hardy D., Manber U., Schwartz M., and Wessels D., "The Harvest Information Discovery and Access System", Internet Research Task Force - Resource Discovery, http://harvest.transarc.com/http://harvest.transarc.com/ .

[4] Wessels D., Claffy K., "ICP and the Squid Web Cache", National Laboratory for Applied Network Research, http://www.nlanr.net/~wessels/Papers/icp-squid.ps.gz.

[5] Wessels D., "The Squid Internet Object Cache", National Laboratory for Applied Network Research, http://squid.nlanr.net/Squid/

[6] mtrace, Xerox PARC, ftp://ftp.parc.xerox.com/pub/net-research/ipmulti/.

[7] R. Atkinson, "Architecture de sécurité pour le protocole Internet", RFC1825, août 1995. *(Obsolète, voir RFC2401)*

[8] R. Atkinson, "En-tête d'authentification IP", RFC18261826 , août 1995. *(Obsolète, voir RFC2402)*

[9] R. Atkinson, "Encapsulation dans IP de charge utile de sécurité (ESP)", RFC18271827 , août 1995. *(Obsolète, voir RFC2406)*

11. Remerciements

Les auteurs tiennent à remercier Paul A Vixie <paul@vix.com> pour les excellents retours fournis sur ce document, Martin Hamilton <martin@mrrl.lut.ac.uk> pour avoir poussé au développement de la diffusion groupée ICP, Eric Rescorla <ekr@terisa.com> et Randall Atkinson <rja@home.net> pour leur assistance sur les questions de sécurité, et tout particulièrement Allyn Romanow pour nous avoir gardés sur la bonne voie.

12. Adresse des auteurs

Duane Wessels
National Laboratory for Applied Network Research
10100 Hopkins Drive
La Jolla, CA 92093 USA
mél : wessels@nlanr.net

K. Claffy
National Laboratory for Applied Network Research
10100 Hopkins Drive
La Jolla, CA 92093 USA
mél : kc@nlanr.net