MOOC
Objectif IPv6 !
vers l'internet nouvelle génération

Document Compagnon[1]

Séquence 2

Le protocole IPv6

Par **- Jean Pierre Rioual** Eurekom / G6

Licence Creative Nommons CC BY-SA 4.0 International

Attribution - Partage dans les Mêmes Conditions 4.0 International (CC BY-SA 4.0)

Avertissement Ce résumé n'indique que certaines des dispositions clé de la licence. Ce n'est pas une licence, il n'a pas de valeur juridique. Vous devez lire attentivement tous les termes et conditions de la licence avant d'utiliser le matériel licencié.

Creative Commons n'est pas un cabinet d'avocat et n'est pas un service de conseil juridique. Distribuer, afficher et faire un lien vers le résumé ou la licence ne constitue pas une relation client-avocat ou tout autre type de relation entre vous et Creative Commons.

Clause C'est un résumé (et non pas un substitut) de la licence.

http://creativecommons.org/licenses/by-sa/4.0/legalcode

Vous êtes autorisé à :

- **Partager** — copier, distribuer et communiquer le matériel par tous moyens et sous tous formats
- **Adapter** — remixer, transformer et créer à partir du matériel
- pour toute utilisation, y compris commerciale.

L'Offrant ne peut retirer les autorisations concédées par la licence tant que vous appliquez les termes de cette licence.

Selon les conditions suivantes :

Attribution — You must give **appropriate credit**, provide a link to the license, and **indicate if changes were made**. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

Partage dans les Mêmes Conditions — Dans le cas où vous effectuez un remix, que vous transformez, ou créez à partir du matériel composant l'Oeuvre originale, vous devez diffuser l'Oeuvre modifiée dans les même conditions, c'est à dire avec **la même licence** avec laquelle l'Oeuvre originale a été diffusée.

No additional restrictions — Vous n'êtes pas autorisé à appliquer des conditions légales ou des **mesures techniques** qui restreindraient légalement autrui à utiliser l'Oeuvre dans les conditions décrites par la licence.

Notes: Vous n'êtes pas dans l'obligation de respecter la licence pour les éléments ou matériel appartenant au domaine public ou dans le cas où l'utilisation que vous souhaitez faire est couverte par une **exception.**

Aucune garantie n'est donnée. Il se peut que la licence ne vous donne pas toutes les permissions nécessaires pour votre utilisation. Par exemple, certains droits comme l**es droits moraux, le droit des données personnelles et le droit à l'image** sont susceptibles de limiter votre utilisation.

Les informations détaillées sont disponibles aux URL suivantes :

- http://creativecommons.org/licenses/by-sa/4.0/deed.fr
- http://fr.wikipedia.org/wiki/Creative_Commons

Tables des activités

Introduction

Le protocole IP a pour objectif de faciliter la transmission de l'information d'un point à un autre du réseau.

- IP est basé sur le modèle datagramme: ce qui signifie que chaque paquet dispose des éléments nécessaires et suffisants au traitement par les équipements indépendamment des paquets traités précédemment.
- IP est le langage commun de tous les équipements de l'Internet, sans besoin de traduction en cours d'acheminement de l'information. IP établit le principe du bout-en-bout: aucun équipement intermédiaire ne perturbe l'information transmise, seuls l'émetteur et le destinataire de l'information sont concernés et actifs.
- La taille des adresses IP source et destination est fixe: ce qui optimise les traitements nécessaires à la transmission dans le réseau.
- L'expérience internet a insufflé et démontré une démultiplication des besoins, et des usages. Il faut désormais répondre aux besoins de communication variés de tous les individus répartis sur la planète, qu'ils soient sédentaires ou en mobilité. Les objets communicants se démultiplient aussi bien en réseau domestique, en entreprise, dans l'industrie, les transports, le milieu médical.
- Il y a quarante ans, le protocole IPv4 a défini des adresses sur 32 bits, mais 4,3 milliards d'adresses s'avèrent aujourd'hui insuffisantes pour les nouveaux usages d'Internet. Tandis que les supports de transmission et les équipements se sont améliorés, le protocole IPv4 a gardé la trace de fonctionnalités historiques nécessaires dans les années 80/90 mais qui n'ont plus cours aujourd'hui.
- Le protocole IPv6 arrive à temps, c'est un retour aux principes qui ont fait le succès d'IP, garantissant efficacité, résilience et des perspectives d'évolution.
- Dans la première séquence vous avez découvert une capacité d'adressage exceptionnelle, dans cette 2° séquence vous allez vous concentrer sur les mécanismes protocolaires. Le fil rouge est l'optimisation du traitement des paquets dans tous les équipements intermédiaires tels que les routeurs, commutateurs de niveau 3, pare-feux. Aux extrémités on retrouve les équipements terminaux tels que stations et serveurs, elles sont responsables de la gestion des en-têtes IP.

Dans cette deuxième séquence du Mooc IPv6 vous aborderez les différents aspects du protocole à travers 5 activités pédagogiques:

- A21: Tout d'abord, vous allez passer en revue le format de l'en-tête des paquets IPv6,
- A22.Ensuite, vous seront exposé les mécanismes d'encapsulation,
- A23: Puis, vous aborderez les principes de routage,
- A24: Ensuite, vous décomposerez les extensions de l'en-tête IP à travers des exemples,
- A25: Enfin les points essentiels sur des tailles de paquets vous seront exposé;

Après avoir approfondi tous ces aspects protocolaires

- A26: Vous pourrez expérimenter IPv6 à travers dans une première séquence de Travaux Pratiques, vous profiterez d'une machine virtuelle intégrant un simulateur réseau très réaliste qui vous permettra de tester, observer et pratiquer, sans déstabiliser la

configuration de votre machine.

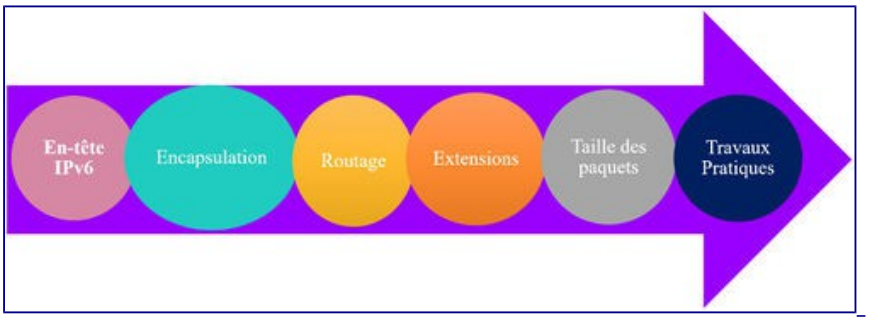

Figure 1: organisation de la séquence.

Activité 21: Le format de l'en-tête IPv6

Introduction

Hormis la modification de la taille des adresses, ce qui conduit à une taille d'en-tête de 40 octets (le double de l'en-tête IPv4 sans les options), le protocole IP a subi un toilettage reprenant l'expérience acquise au fil d'une quarantaine d'années avec IPv4. Le format des en-têtes IPv6 est simplifié et permet aux routeurs de meilleures performances dans leurs traitements:

- L'en-tête ne contient plus le champ checksum, qui devait être ajusté par chaque r¬outeur en raison entre autres de la décrémentation du champ durée de vie. Par contre, pour éviter qu'un paquet dont le contenu est erroné -- en particulier sur l'adresse de destination -- ne se glisse dans une autre communication, tous les protocoles de niveau supérieur doivent mettre en œuvre un mécanisme de checksum de bout en bout incluant un pseudo-en-tête qui prend en compte les adresses source et destination. Le checksum d'UDP, facultatif pour IPv4, devient ainsi obligatoire. Pour ICMPv6, le checksum intègre le pseudo-en-tête, alors que pour ICMPv4, il ne portait que sur le message ICMP.
- La taille des en-têtes étant fixe, le routeur peut facilement déterminer où commence la zone de données utiles.
- Les options ont été retirées de l'en-tête et remplacées par de nouveaux en-têtes appelés extensions qui peuvent être facilement ignorées par les routeurs intermédiaires.
- Les champs sont alignés sur des mots de 64 bits, ce qui optimise leur traitement, surtout avec les nouvelles architectures à 64 bits.
- La taille minimale des MTU: Maximum Transmission Unit est de 1 280 octets. Le choix de 1 280 comme MTU minimal en IPv6 permet la mise en tunnel de paquets IPv6. En effet, la taille de 1 500 octets est généralement admise car elle correspond à la valeur imposée par Ethernet. La majorité des autres réseaux offrent une taille supérieure. Pour les réseaux ne le permettant pas, une couche d'adaptation (comme avec les couches d'adaptation AAL5 d'ATM [RFC 2492] ou 6LoWPAN avec les réseaux de capteurs (comme IEEE 802.15.4 [RFC 4944]) devra être mise en œuvre pour pouvoir transporter les paquets IPv6.
- La fonction de fragmentation a été retirée des routeurs. Les champs qui s'y reportent (identification, drapeau, place du fragment) ont été supprimés. Normalement les algorithmes de découverte du PMTU (Path MTU) évitent d'avoir recours à la fragmentation. Si celle-ci s'avère nécessaire, une extension est prévue.

L'idée est de retirer du cœur de réseau les traitements compliqués. Les routeurs ne font que retransmettre les paquets vers la destination, les autres traitements (fragmentation, ...) seront faits par l'émetteur du paquet.

Format de l'en-tête du paquet IPv6

Le format d'en-tête du paquet IPv6 est spécifié par le RFC 2460 page 4. Cette en-tête avec les champs le composant est représenté par la figure 1. Comme indiqué précédemment, l'en-tête IPv6 est de taille fixe et se compose de mots de 64 bits. De manière similaire à IPv4, l'en-tête

se compose de 5 mots. La taille des mots est doublée dans le cas d'IPv6. La taille de l'en-tête IPv6 est ainsi de 40 octets.

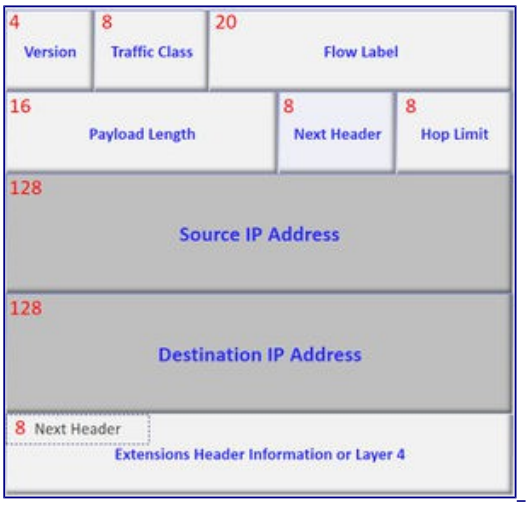

Figure 1: Format d'un paquet IPv6

Valeurs des champs de l'en-tête

Version

Le champ version est le seul champ qui occupe la même place dans le paquet IPv6 et dans le paquet IPv4. Sa valeur est 6. Le numéro de version 5 avait déjà été attribué au protocole Stream qui finalement n'a pas eu le succès attendu (RFC 1190 , RFC 1819).

Classe de trafic

Dans la version standardisée par le RFC 2460 un champ classe de trafic sur 8 bits permet la différenciation de services conformément aux spécifications du RFC 2474 .

Le champ classe de trafic est aussi appelé dans les paquets IPv4 octet DiffServ (DS), il prend la place du champ ToS, initialement défini dans la spécification d'IPv4 (cf. figure 2). Le champ DS est découpé en deux parties. Le sous-champ DSCP (*DiffServ Code Point*) contient les valeurs des différents comportements. Les deux derniers bits du champ notés ECN (*Explicit congestion Notification*) servent aux routeurs à reporter un risque de congestion en combinaison avec l'algorithme RED (*Random Early Detection*). Le codage des 2 bits ECN est décrit à la page 6 du RFC 6040

Figure 2: Format de l'octet classe de trafic.

L'Internet différencié permet aux fournisseurs d'accès de gérer différemment les congestions qui surviennent dans le réseau. Sans différenciation, les paquets ont la même probabilité de rejet. Avec la différenciation, plusieurs classes de trafic seront définies. Les paquets appartenant aux

classes les plus élevées ont une probabilité de rejet plus faible. Bien entendu pour que l'introduction de telles classes de service soit efficace, il faut introduire une gestion des ressources différente pour chacune des classes et des mécanismes de contrôle pour vérifier que les flux des utilisateurs n'utilisent pas que les classes les plus élevées ou qu'ils ne dépassent leur contrat. Par exemple un client peut établir un contrat de niveau de services appelé SLA (*Service Level Agreement*) avec son fournisseur d'accès .

L'intérêt principal de la différenciation de services est qu'elle ne casse pas le modèle initial de l'Internet (version 4 ou version 6). Les flux sont toujours traités en *Best Effort* même si certains sont plus *Best* que d'autres. Il n'y a aucune garantie qu'un trafic d'une classe de service haute arrive à destination, mais la probabilité est plus importante. L'autre intérêt des classes de service vient de la possibilité d'agrégation des flux. La classe d'appartenance est indiquée dans l'en-tête du paquet. Les applications peuvent marquer les paquets en fonction de paramètres locaux (flux multimédia, flux interactif, trafic priorisé, ...). Le fournisseur d'accès qui récupère le trafic n'a plus à se préoccuper des applicatifs, il vérifie que le trafic d'une classe ne dépasse pas le contrat préalablement établi.

Dans le cœur du réseau, les routeurs prennent en compte les différentes classes. Le fournisseur d'accès devra également passer des accords avec les autres opérateurs pour pouvoir faire transiter les flux avec un traitement approprié. Cet aspect de dimensionnement de réseau et de négociation d'accords d'échange est au coeur du métier d'opérateur.

Le tableau 1 présente les différentes valeurs définies pour le champ DSCP (*Differential Service Code Point*). Les valeurs sont présentées en format binaire avec les 6 bits les plus significatifs de l'octet Traffic Class, puis leur conversion en décimal, leur nommage, la probabilité d'écartement, et l'équivalence avec les anciennes valeurs du champ TOS de l'IPv4:

DSCP Value	Decimal Value	Meaning	Drop Probability	Equivalent Precedence Value
101 110	46	EF Expedited Forwarding	N/A	101 – Critical
101 100	44	Voice Admit	N/A	101 – Critical (RFC 5865)
000 000	00	Best Effort / Default	N/A	000 – Routine
001 010	10	AF11 Assured Forwarding	Low	001 – Priority
001 100	12	AF12	Medium	001 – Priority
001 110	14	AF13	High	001 – Priority
010 010	18	AF21	Low	010 – Immediate
010 100	20	AF22	Medium	010 – Immediate
010 110	22	AF23	High	010 – Immediate
011 010	26	AF31	Low	011 – Flash
011 100	28	AF32	Medium	011 – Flash
011 110	30	AF33	High	011 – Flash
100 010	34	AF41	Low	100 – Flash Override
100 100	36	AF42	Medium	100 – Flash Override
100 110	38	AF43	High	100 – Flash Override
001 000	08	CS1 Class Selector		1 – Priority
010 000	16	CS2		2 – Immediate
011 000	24	CS3		3 – Flash
100 000	32	CS4		4 – Flash Override
101 000	40	CS5		5 – Critic /ECP
110 000	48	CS6		6 – Internetwork Control
111 000	56	CS7		7 – Network Control

Tableau 1: Format du champ DSCP.

Pour l'instant deux types de comportement sont standardisés:

- Assured Forwarding [RFC 2597]: Ce comportement définit quatre classes de services et trois priorités suivant que l'utilisateur respecte son contrat, le dépasse légèrement ou est largement en dehors. Les classes sont donc choisies par l'utilisateur et restent les mêmes tout au long du trajet dans le réseau. La priorité, par contre, peut être modifiée dans le réseau par les opérateurs en fonction du respect ou non des contrats.Par exemple pour la classe AF n°2 on dispose des 3 priorités suivantes: AF21, AF22, AF23, plus le chiffre est élevé, plus la priorité est faible, c'est à dire qu'en cas de saturation de cette classe de service, les paquets AF23 seront écartés avant AF22, puis AF21.
- Expedited Forwarding [RFC 2598]: Ce comportement est comparable à un circuit à débit constant réservé dans le réseau. Le trafic est mis en forme à l'entrée du réseau, en retardant l'émission des paquets qui sont hors contrat. En plus de ces comportements, l'octet DS a gardé, pour des raisons de compatibilité avec les équipements existants, les valeurs du bit ToS qui étaient le plus fréquemment utilisées. La valeur est 0xEF (1011 1000 en binaire, et en tenant compte des 6 bits de poids forts: 46 en décimal).
- Voice Admit: cette autre valeur a été par la suite proposée dans le RFC 5865 pour affiner le traitement de flux temps réels de différentes natures (Voix, Video, Signalisation Temps réel…). Et une autre particularité, la valeur 0xE0 (1110 0000 en binaire, et en tenant compte des 6 bits de poids forts: 56 en décimal) correspond à la classe de contrôle du réseau (*Network Control*). Elle est utilisée dans des mises en oeuvre d'IPv6 pour l'émission de certains paquets ICMPv6.

Identificateur de flux

Ce champ introduit dans le RFC 2460 puis spécifié en détail dans le RFC 6437 contient un numéro unique choisi par la source, qui a pour but de faciliter le travail des routeurs et la mise en oeuvre des fonctions de qualité de service comme RSVP. Cet indicateur peut être considéré comme une marque à un contexte dans le routeur. Le routeur peut alors faire un traitement particulier: choix d'une route, traitement en "temps-réel" de l'information.

Avec IPv4, certains routeurs, pour optimiser le traitement, se basent sur les valeurs de cinq champs pour construire un contexte: adresses de la source et de destination, numéros de port de la source et de destination et protocole. Ce contexte sert à router plus rapidement les paquets puisqu'il évite de consulter les tables de routage pour chaque paquet. Ce contexte est détruit après une période d'inactivité.

Avec IPv6, cette technique est officialisée. Le champ identificateur de flux peut être rempli avec une valeur aléatoire qui servira à référencer le contexte. La source gardera cette valeur pour tous les paquets qu'elle émettra pour cette application et cette destination. Le traitement est optimisé puisque le routeur n'a plus à consulter cinq champs pour déterminer l'appartenance d'un paquet. De plus si une extension de confidentialité est utilisée, les informations concernant les numéros de port sont masquées aux routeurs intermédiaires.

L'utilisation de ce champ a été rendue confuse car Cisco dans le cadre du Tag Switching a

proposé de l'utiliser pour augmenter la vitesse de commutation des paquets. Cette proposition consiste à ne garantir l'unicité de l'identificateur de flux que sur un lien. Le routeur possède dans sa mémoire une table de correspondance qui permet, en fonction du lien d'arrivée et du numéro d'identificateur de flux, de déterminer le lien de sortie et la nouvelle valeur de l'identificateur. Cette proposition se rapproche énormément des techniques utilisées dans les circuits virtuels (ATM, Frame Relay, X.25...).

Le groupe de travail MPLS (*Multi Protocol Label Switching*) a intégré les travaux sur le Tag Switching et a précisé la manière dont la commutation des paquets pourra être faite. L'identificateur de flux d'IPv6 n'est plus utilisé, mais un en-tête spécifique est introduit entre l'encapsulation de niveau 2 et celle de niveau 3. L'identificateur de flux n'a plus à être modifié en cours de transmission. Cette évolution clarifie l'utilisation du protocole RSVP (*Reservation Protocol*) qui peut se baser sur cette valeur, identique tout au long du chemin, pour identifier un flux.

En fait, l'utilisation de l'étiquette de flux est très floue, les micro-flux, c'est-à-dire de flux applicatifs, ne sont pas vus dans le coeur du réseau pour des raisons de facteur d'échelle, de plus MPLS a repris la notion de routage spécifique en fonction d'une étiquette. Pour l'instant ce champ peut être vu comme réservé et son utilisation pourra être mieux spécifiée dans le futur.

Longueur des données utiles (payload)

Contrairement à IPv4, ce champ, sur deux octets, ne contient que la taille des données utiles, sans prendre en compte la longueur de l'en-tête. Pour des paquets dont la taille des données serait supérieure à 65 535 ce champ vaut 0 et l'option jumbogramme de l'extension de "proche-en-proche" est utilisée (cf. Jumbogramme) (type 194 ou 0xc2, RFC 2675). Cette option est utilisée quand le champ longueur des données du paquet IPv6 n'est pas suffisant pour coder la taille du paquet. Cette extension est essentiellement prévue pour la transmission à grand débit entre deux équipements. Si l'option jumbogramme est utilisée, le champ longueur des données utiles dans l'en-tête IPv6 vaut 0. Noter que le type commence par la séquence binaire 11, ce qui permet au routeur ne traitant pas les jumbogrammes d'en informer la source. Celle-ci pourra réémettre l'information sans utiliser cette option.).

En-tête suivant

Ce champ a une fonction similaire au champ protocole du paquet IPv4. Il identifie le prochain en-tête. Il peut s'agir d'un protocole (de niveau supérieur ICMP, UDP, TCP...) ou de la désignation d'extensions (cf. tableau 2). Les extensions contiennent aussi ce champ pour permettre un chaînage.

Codage	Description
0	Proche en proche
4	IPv4
6	TCP
17	UDP
41	IPv6
43	Routage
44	Fragmentation
50	ESP confidentialité
51	AH Authentification
58	ICMPv6
59	IPv6 No Next
60	Destination
135	Mobilité
136	UDP-lite
140	Shim6
194	Jumbogramme

Tableau 2: Valeurs du champ en-tête suivant.

Nombre de sauts

Il est décrémenté à chaque nœud traversé. Un datagramme retransmis par un routeur est rejeté avec l'émission d'un message d'erreur ICMPv6 vers la source si la valeur après décrémentation atteint 0. Dans IPv4 ce champ est appelé durée de vie (ou TTL Time To Live). Sa vocation initiale est d'indiquer, en secondes, la durée maximale durant laquelle un paquet peut rester dans le réseau. En pratique, les paquets ne restent que quelques millisecondes dans les routeurs, et donc la décrémentation est arrondie à 1. Par contre, pour une liaison plus lente la décrémentation de ce champ peut être supérieure à 1. Dans IPv6, comme il s'agit d'un nombre de sauts, la décrémentation est toujours de 1. La valeur initiale de ce champ devrait être donnée dans un document annexe de l'IANA (http://www.iana.org/) ce qui permettrait de la modifier en fonction de l'évolution de la topologie du réseau. La valeur n'est pas encore officiellement attribuée, mais certaines implantations prennent actuellement la valeur conseillée pour IPv4: 64.

La valeur par défaut peut être dynamiquement attribuée aux équipements du réseau par les annonce des routeurs en configuration automatique, une modification de ce paramètre sera donc relativement simple quand la limite actuelle sera atteinte. On peut noter une limitation, puisque ce champ codé sur 8 bits n'autorise la traversée que de 255 routeurs. En réalité, dans l'Internet actuel, le nombre maximal de routeurs traversés est d'une quarantaine, ce qui laisse une bonne marge pour l'évolution du réseau.

Adresses source et destination

Traitées précédemment dans la séquence 1.

Extensions ou couches de niveau supérieur

En IPv4 les options étaient utilisées pour fournir des informations supplémentaires nécessaire à un traitement particulier des routeurs, en IPv6 les extensions ont compensé ce dispositif et permis d'optimiser le traitement des équipements intermédiaires.

La figure 3 montre la souplesse avec laquelle plusieurs extensions peuvent être chaînées. Chaque extension contient dans son en-tête un champ en-tête suivant et longueur. Le premier paquet ne contient pas d'extension, le champ en-tête suivant pointe sur TCP. Le second paquet contient une extension de routage qui pointe sur TCP. Dans le dernier paquet, une extension de fragmentation est ajoutée après celle de routage.

Si cet enchaînement d'extension offre beaucoup plus de souplesse que les options d'IPv4, il rend difficile la lecture des numéros de port, il faut en effet lire tout l'enchaînement d'extension pour arriver au protocole de niveau 4. Ceci a servi de justification au l'identificateur de flux qui permettait de refléter au niveau 3 un flux particulier et évitait de dérouler l'enchaînement. Bien entendu, les pare-feux devront vérifier les numéros de ports.

Les extensions peuvent être vues comme un protocole 3.5 (entre la couche 3 et la couche 4). En effet, à part l'extension de proche-en-proche, qui est traitée par tous les routeurs traversés, les autres extensions ne sont traitées que par le destinataire du paquet (i.e. celui spécifié dans le champ adresse de destination du paquet IPv6).

Si d'un point de vue théorique les extensions sont supérieurs aux options d'IPv4, dans la réalité très peu sont utilisées à grande échelle et restent du domaine de la recherche.

Nous reviendrons plus sur les détails des extensions dans l'activité 24.

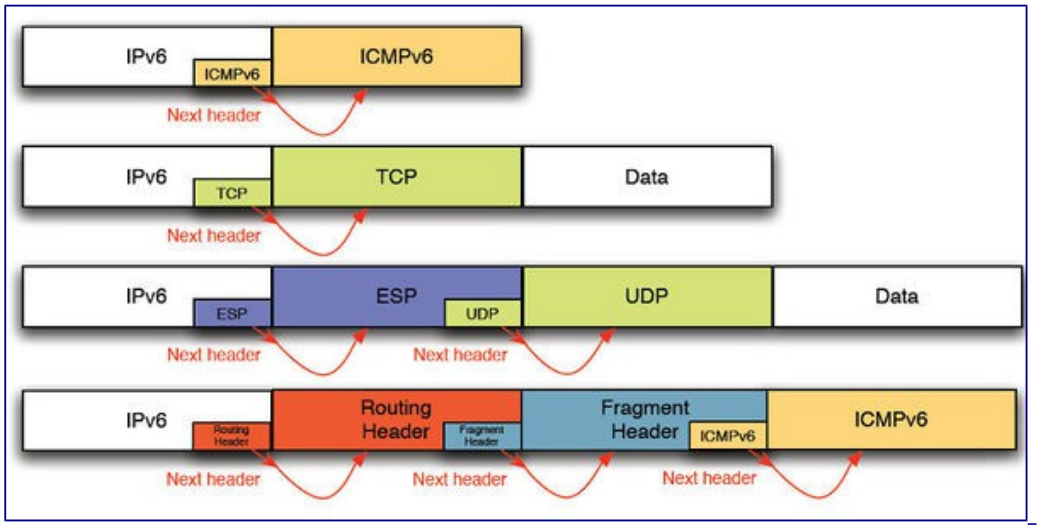

Figure 3: Enchaînement d'extensions IPv6.

Références bibliographiques

Vous pouvez approfondir vos connaissances en consultant les liens suivants:

- RFC 1190 : Experimental Internet Stream Protocol Version 2 (ST-II)
- RFC 1819 : Internet Stream Protocol Version 2 (ST2), Protocol Specification

- RFC 2460 : Internet Protocol, Version 6 (IPv6) Specification (Analyse)
- RFC 2492 : IPv6 over ATM Networks (cf page 2)
- RFC 2474 : Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers
- RFC 2597 : An Expedited Forwarding PHB
- RFC 2598 : Assured Forwarding PHB Group
- RFC 4594 : Configuration Guidelines for DiffServ Service Classes
- RFC 5865 : A Differentiated Services Code Point (DSCP) for Capacity-Admitted Traffic
- RFC 6040 : Tunnelling of Explicit Congestion Notification
- RFC 6437 : IPv6 Flow Label Specification
- RFC 6438 : Flow Label for Tunnel ECMP/LAG (cf page-4)
- RFC 4944 : Transmission of IPv6 Packets over IEEE 802.15.4 Networks
- RFC 6282 : Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks
- RFC 6775 : Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)

Pour aller plus loin

Cisco (2006) White paper. IPv6 Extension Headers Review and Considerations

Activité 22: Les mécanismes d'encapsulation

Introduction

La représentation de l'encapsulation de protocoles utilise le principe de l'empilement des couches représentatives des traitements nécessaires à effectuer dans les différents composants d'un réseau. Ces traitements affecteront toutes les couches dans les équipements d'extrémités, et certaines seulement pour les équipements réalisant le relais des échanges sur le réseau de transport.

Représentation de l'encapsulation

L'organisme ISO a défini le modèle OSI par une représentation en 7 couches représentées du niveau Physique jusqu'au niveau Application, le modèle DOD TCP/IP a simplifié cette représentation.

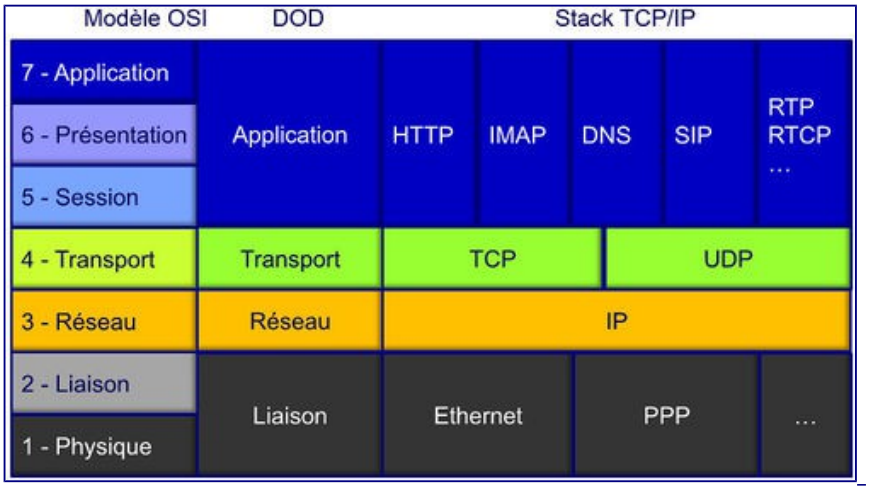

Figure 1: Comparaison modèle ISO / Stack TCP/IP.

Pour simplifier l'organisation, nous pouvons considérer par exemple que pour une une configuration d'un poste de travail, la carte réseau réalise les fonctions de niveau Physique et Liaison, que le traitement des couches Réseau et Transport est réalisée par les couches intermédiaires installées dans le système d'exploitation, et que le reste du système avec les programmes applicatifs gère les couches Session, Présentation et Application.

Traitement des couches basses

La méthode de transport d'un datagramme IPv6 entre deux machines directement reliées entre elles par un lien physique est le même que pour IPv4: le datagramme est tout d'abord routé vers une interface d'émission qui l'encapsule dans une trame (PDU de niveau 2 dans le modèle de référence de l'OSI); cette trame est transmise sur le lien vers l'adresse physique de la machine destination (cette adresse sur un lien sera appelée Adresse MAC dans la suite); la machine destination reçoit la trame sur son interface, la décapsule et la traite.

Les différences avec IPv4 sont:

- Sur le support Ethernet RFC 2464 précise que le code protocole encapsulé de la trame

est différent. Par exemple, pour les réseaux à diffusion, le code est 0x86DD alors que pour IPv4 le code est 0x0800. À l'origine, il était prévu de garder le même code et d'assurer l'aiguillage entre IPv4 et IPv6 en utilisant le champ version du paquet. Mais certains équipements ne vérifient pas la valeur de ce champ et auraient eu un comportement incontrôlable en essayant de traiter un paquet IPv6 comme un paquet IPv4.

- Le calcul de l'adresse MAC destination change. Par exemple sur un réseau à diffusion le calcul est fait en IPv4 par le protocole ARP, alors qu'en IPv6 on utilise le protocole de découverte de voisins .
- La taille minimale d'une trame est passée à 1 280 octets; ceci peut forcer certains protocoles à utiliser plusieurs trames par datagramme IPv6.
- Enfin, certains protocoles ont des parties propres à IPv4. Ces parties doivent être modifiés. C'est le cas des protocoles de contrôle et de compression de PPP.

Couche physique

Commençons par la couche Physique qui est à la base de l'édifice de ce modèle. Les spécifications de cette couche dépendent du support lui-même. Nous devons gérer la transmission des informations binaires issue du codage des trames et des paquets sur un support cuivre, optique ou sans fils; d'où la nécessité d'adaptation aux caractéristiques des composants (câbles, connecteurs ou antennes) et d'une méthode appropriée de codage des données (représentation physique des données).

La représentation binaire utilisée dépend du support, sur du cuivre on utilise des variations d'impulsions électriques, en optique ce sont des variations lumineuses sur une ou plusieurs longueurs d'ondes, en sans fils ce sont généralement des signaux radios, laser ou infrarouge. La couche Physique coordonne le débit et la synchronisation de l'émetteur et du récepteur réseau, tout en tentant de garantir la transparence et l'intégrité d'un flux d'information binaire, sans notion d'interprétation du contenu.

Hélas cette couche est fréquemment soumise à différentes perturbations issues d'un monde extérieur au canal de transmission, radiations électromagnétiques, microcoupures ou altérations des signaux par différents facteurs. Les coupleurs intégrés dans les cartes réseaux réalisent les fonctions nécessaires et utiles au niveau Physique, et un dispose d'un indicateur de qualité de la transmission avec le calcul de CRC: Contrôle de Redondance Cyclique, appelé Checksum.

Couche liaison

Le rôle de la couche liaison est en autres de transformer la couche physique en une liaison a priori exempte d'erreurs de transmission pour la couche réseau. De plus elle permet d'occuper le lien en fonction des besoins d'émission ou de récupérer toutes les transmissions fiables réceptionnées. Etant donné que pour la couche physique, les données n'ont aucune signification particulière. La couche liaison de données doit donc être capable d'écarter le trafic nécessaire à la synchronisation, et de reconnaître les débuts et fin de trames. Cette couche écarte les trames en cas de réception erronée, comme par exemple non-respect du format, ou bien en cas de problème sur la ligne de transmission; la vérification du champ CRC aide à faire

ce tri. Cette couche intègre également une fonction de contrôle de flux pour éviter l'engorgement d'un récepteur incapable de suivre un rythme imposé.

L'unité de donnée de protocole de la couche liaison de données est la trame (LPDU: Link Protocol Data Unit) qui est composée de plusieurs champs permettant d'identifier l'origine des échanges, le rôle de la trame, et le contenu de l'enveloppe, ainsi en fin de trame le champ CRC, le tout étant encadré par une séquence particulière de codage début et fin de trame.

Si nous prenons l'exemple de la trame Ethernet, un délai inter-trame minimum de 96 intervalles de temps est spécifié comme silence sur un support cuivre, alors que sur un support optique tout silence est comblé par la transmission d'un ou plusieurs symboles particuliers «idle», une parfaite synchronisation est alors maintenue entre les extrémités du lien optique.

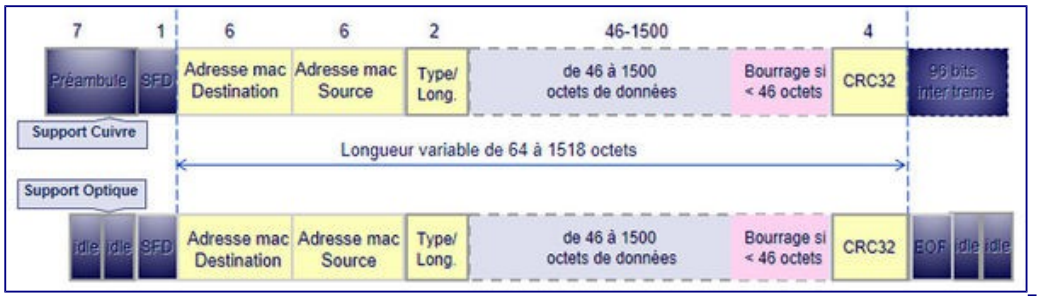

Figure 2: Format de la trame Ethernet.

Une fois que l'arrivée d'une trame Ethernet est détectée par le coupleur, les premiers champs immédiatement accessibles correspondent aux adresses mac Destination et Source, puis soit au champ Longueur dans le cas d'une encapsulation au standard 802.3, ou bien au champ EtherType dans le cas d'une encapsulation avec le standard Ethernet original. Ensuite l'enveloppe de la trame transporte les données, qui correspondent aux paquets IPv6 dès lors que le champ EtherType=0x86dd, vient ensuite le champ CRC codé sur 32 bits. Dans le cas d'encapsulation 802.1Q, d'autres champs permettent la reconnaissance du numéro de vlan et du niveau de priorité défini dans le standard 802.1p.

Un des éléments particulièrement importants est la capacité de transport de de la trame, dans l'exemple ci-dessus nous voyons que la trame Ethernet traditionnelle dispose d'une enveloppe qui autorise le transport de 1500 octets maximum, MTU=1500 (MTU=Maximum Transmit Unit).

D'autres formats de trames permettent des échanges plus ou moins importants, citons quelques MTU:

- PPPoE = 1492
- PPPoA = 1468
- MPLS = 1500-65535
- 6LoWPAN = 81
- Ethernet Jumboframe = 9000

Couches intermédiaires

Couche réseau

Etant donné que la taille minimum de l'en-tête IPv6 est de 40 octets, le MTU résiduel d'une trame Ethernet classique est de 1500-40=1460 octets Sachant que ces 1460 de données seront probablement encore amputées d'en-têtes de niveau transport, par exemple 20 octets minimum pour TCP et 8 octets pour UDP. A ce niveau rappelons qu'aucun champ CRC n'a été retenu, car nous en avons déjà au niveau liaison, et les couches supérieures vont s'en charger.

Couches Transport

UDP et TCP

Les modifications apportées aux protocoles de niveau 4 UDP et TCP sont minimes. L'un des pré-requis à la mise en œuvre d'IPv6 était de laisser en l'état aussi bien TCP (Transmission Control Protocol) qu'UDP (User Datagram Protocol). Ces protocoles de transport sont utilisés par la très grande majorité des applications réseau et l'absence de modification facilitera grandement le passage de IPv4 à IPv6.

La principale modification à ces protocoles concerne le checksum. Comme il a été précisé Checksum au niveau transport, il a été adapté au format de paquet IPv6 et englobe le pseudo-en-tête. De plus, pour UDP, le checksum qui était facultatif en IPv4, devient obligatoire.

Un autre changement au niveau des protocoles de niveau 4 concerne la prise en compte de l'option jumbogramme de l'extension proche-en-proche. Le RFC 2675 définit le comportement d'UDP et de TCP quand les jumbogrammes sont utilisés. En effet, les en-têtes de ces messages contiennent eux aussi un champ longueur codé sur 16 bits et par conséquent insuffisant pour coder la longueur du jumbogramme:

- Pour le protocole UDP, si la longueur des données excède 65 535 octets, le champ longueur est mis à 0. Le récepteur détermine la longueur des données par la connaissance de la taille dans l'option jumbogramme.
- Le protocole TCP pose plus de problèmes. En effet, bien que les messages TCP ne contiennent pas de champ longueur, plusieurs compteurs sont codés sur 16 bits.
- Le champ longueur de la fenêtre de réception ne pose pas de problème depuis que le RFC 1323 a défini l'option TCP window scale qui donne le facteur multiplicatif qui doit être appliqué à ce champ.
- À l'ouverture de connexion, la taille maximale des segments (MSS) est négociée. Le RFC 2675 précise que si cette taille doit être supérieure à 65 535, la valeur 65 535 est envoyée et le récepteur prend en compte la longueur déterminée par l'algorithme de découverte du MTU.

Pour l'envoi de données urgentes avec TCP, on utilise un bit spécifique de l'en-tête (bit URG) ainsi que le champ "pointeur urgent". Ce dernier sert à référencer la fin des données à traiter de manière particulière. Trois cas peuvent se présenter:

- Le premier, qui est identique à IPv4, est celui où le pointeur indique une position de moins de 65 535.
- Le second se produit lorsque le déplacement est supérieur à 65 535 et supérieur ou égal à la taille des données TCP envoyées. Cette fois-ci, on place la valeur 65 535 dans le champ "pointeur urgent" et on continue le traitement normal des paquets TCP.
- Le dernier cas intervient quand le pointeur indique un déplacement de plus de 65 535 qui est inférieur à la taille des données TCP. Un premier paquet est alors envoyé, dans lequel on met la valeur 65 535 dans le champ "pointeur urgent". L'important est de choisir une taille de paquet de manière à ce que le déplacement dans le second paquet, pour indiquer la fin des données urgentes, soit inférieur à 65 535.

Il existe d'autres propositions pour faire évoluer TCP. Il faut remarquer que le travail n'est pas de même ampleur que pour IP. En effet, TCP est un protocole de bout-en-bout, la transition vers une nouvelle génération du protocole peut se faire par négociation entre les deux extrémités. Pour IP, tous les routeurs intermédiaires doivent prendre en compte les modifications.

UDP-lite

UDP-lite permet de remonter aux couches supérieures des données erronées pendant leur transport. Si dans un environnement informatique, une erreur peut avoir des conséquences relativement graves quant à l'intégrité des données et il est normal de rejeter ces paquets, or, la plupart des décodeurs de flux multimédias sont capables de supporter un certain nombre d'erreurs binaires dans un flux de données. Pour améliorer la qualité perçue par l'utilisateur, il est donc préférable d'accepter des paquets erronés plutôt que de rejeter un bloc complet d'information.

En IPv4, l'utilisation du checksum UDP étant optionnelle (la valeur 0 indique que le checksum n'est pas calculé), UDP peut être utilisé pour transporter des flux multimédias. Avec IPv6, l'utilisation du checksum a été rendue obligatoire puisque le niveau 3 n'en possède pas. Pour éviter qu'un paquet comportant des erreurs ne puisse pas être remonté aux couches supérieures, le protocole UDP-lite a été défini RFC 3828 . Les modifications sont minimes par rapport à UDP. Le format de la trame reste le même, seule la sémantique du champ longueur est changée. Avec UDP, ce champ est inutile puisqu'il est facilement déduit du champ longueur de l'en-tête IP. UDP-lite le transforme en champ couverture du checksum. Si la longueur est 0, UDP-lite considère que tout le checksum couvre tout le paquet. La valeur 8 indique que seul l'en-tête UDP est protégé par le checksum (ainsi qu'une partie de l'en-tête IP grâce au pseudo-header). Les valeurs comprises entre 1 et 7 sont interdites car le checksum UDP-lite doit toujours couvrir l'en-tête. Une valeur supérieure à 8 indique qu'une partie des données sont protégées. Si la couverture est égale à la longueur du message on se retrouve dans un cas compatible avec UDP.

SCTP

Le protocole SCTP (*Stream Control Transmission Protocol*) RFC 2960 est fortement lié au protocole IPv6. SCTP est un protocole de niveau 4 initialement conçu pour transporter des informations de signalisation. La fiabilité est donc un prérequis important et la gestion de la

multi-domiciliation est prise en compte. L'idée est de permettre aux deux équipements terminaux d'échanger à l'initialisation de la connexion (appelée dans le standard association), l'ensemble de leurs adresses IPv4 et IPv6. Chaque équipement choisi une adresse privilégiée pour émettre les données vers l'autre extrémité et surveille périodiquement l'accessibilité des autres adresses. Si l'équipement n'est plus accessible par l'adresse principale, une adresse secondaire sera choisie.

SCTP permet une transition douce d'IPv4 vers IPv6 puisque l'application n'a plus à se préoccuper de la gestion des adresses. Si les deux entités possèdent une adresse IPv6, celle-ci sera privilégiée. De plus, SCTP peut servir de brique de base à la gestion de la multi-domiciliation IPv6. En effet, avec TCP une connexion est identifiée par ses adresses. Si une adresse n'est plus accessible, le fait d'en changer peut conduire à la coupure de la connexion. Il faut avoir recours à des superfuges, comme la mobilité IP pour maintenir la connexion. SCTP brise ce lien entre la localisation de l'équipement et l'identification des associations.

Rôle du checksum

Parmi les différences existant entre les datagrammes IPv4 et IPv6, il y a la disparition du checksum dans les en-têtes IP. Cette somme de contrôle était utilisée pour vérifier la validité de l'en-tête du paquet traité. En IPv4, il est nécessaire de la vérifier et de l'ajuster lors de chaque retransmission par un routeur, ce qui entraîne une augmentation du temps de traitement du paquet. Cette somme ne vérifie que l'en-tête IPv4, pas le reste du paquet. Aujourd'hui les supports physiques sont de meilleure qualité et savent détecter les erreurs (par exemple, Ethernet a toujours calculé sa propre somme de contrôle; PPP, qui a presque partout remplacé SLIP, possède un CRC). L'intérêt de la somme de contrôle a diminué et ce champ a été supprimé de l'en-tête IPv6.

Le checksum sur l'en-tête IPv6 n'existant plus, il faut quand même se prémunir des erreurs de transmission. En particulier, une erreur sur l'adresse de destination va faire router un paquet dans une mauvaise direction. Le destinataire doit donc vérifier que les informations d'en-tête IP sont incorrectes pour éliminer ces paquets. Dans les mises en oeuvre des piles de protocoles Internet, les entités de niveau transport remplissent certains champs du niveau réseau. Il a donc été décidé que tous les protocoles au-dessus d'IPv6 devaient utiliser une somme de contrôle intégrant à la fois les données et les informations de l'en-tête IPv6. La notion de pseudo-en-tête dérive de cette conception. Pour un protocole comme TCP qui possède une somme de contrôle, cela signifie modifier le calcul de cette somme. Pour un protocole comme UDP qui possède une somme de contrôle facultative, cela signifie modifier le calcul de cette somme et le rendre obligatoire.

IPv6 a unifié la méthode de calcul des différentes sommes de contrôle. Celle-ci est calculée sur l'ensemble formé de la concaténation d'un pseudo-en-tête (cf. Champ du pseudo-en-tête) et du paquet du protocole concerné. L'algorithme de calcul du checksum est celui utilisé en IPv4. Il est très simple à mettre en œuvre et ne demande pas d'opérations compliquées. Il s'agit de faire la somme en complément à 1 des mots de 16 bits du pseudo-en-tête, de l'en-tête du protocole de transport, et des données, puis de prendre le complément à 1 du résultat.

Il faut noter que les informations contenues dans le pseudo-en-tête ne seront pas émises telles quelles sur le réseau. Le champ "en-tête suivant" du pseudo-en-tête ne reflète pas celui qui sera émis dans les paquets puisque les extensions ne sont pas prises en compte dans le calcul du checksum. Ainsi, si l'extension de routage est mise en œuvre, l'adresse de la destination est celle du dernier équipement. De même le champ longueur est sur 32 bits pour contenir la valeur de l'option jumbogramme, si celle-ci est présente.

Activité 23: Les principes du routage IPv6

Introduction

Le routage IPv6 est quasiment identique au routage IPv4 sous CIDR (Classless Inter-Domain Routing, routage inter-domaine sans classe). La seule différence est la taille des adresses qui sont de 128 bits dans IPv6 au lieu de 32 bits dans IPv4. Beaucoup d'équipements proposent en standard l'intégration du routage statique, et d'autres plus évolués ou avec des extensions, il devient possible d'utiliser des algorithmes de routage dynamique comme OSPF, RIP et IS-IS.

En IPv6, cela ne diffère pas tellement de l'IPv4. La syntaxe est globalement la même, à la différence qu'en IPv6 nous n'avons pas d'adresse "réseau". C'est donc ce qu'on appelle le Préfixe IPv6 qui servira de référence pour notre table de routage.

Cette activité a pour objectif de montrer l'impact d'IPv6 sur les protocoles de routage. Il ne sera pas détaillé ici le fonctionnement de tel ou tel protocole, mais plutôt les changements qui ont été nécessaires afin de prendre en compte la technologie IPv6 dans les protocoles de routage existants pour IPv4. Ces changements sont essentiellement liés à la prise en compte du nouveau format de l'adresse IPv6 ainsi qu'à l'ajout d'une nouvelle table de routage dédiée à IPv6.

Les différents types de routage sont passés en revue: routage statique, routage interne et routage externe. A l'issue du chapitre, on constatera qu'IPv6 est maintenant bien intégré dans ces protocoles et que cette évolution a eu un impact très faible pour l'utilisateur final.

Les algorithmes de routage n'ont pas changé avec IPv6. Les travaux en cours consistent principalement à les adapter au nouveau format de l'adresse IP. Ces protocoles de routage profitent des propriétés maintenant incluses dans la nouvelle version du protocole IPv6 comme l'authentification ou le multicast. Une conséquence de l'apparition du routage IPv6 est que les équipements doivent alors prendre en compte les deux piles de protocoles, IPv4 et IPv6. Cela doit être pris en considération lors de l'activation des protocoles de routage. En particulier, il faut prêter attention à la congruence des topologies IPv4 et IPv6.

Routage Statique

Adresse locale

Pour atteindre une destination la machine cherche à vérifier si l'adresse cible est accessible directement en utilisant les interfaces connectées, plusieurs cas sont possibles:

- Interface de loopback, cas particulier d'une interface logique toujours connectée, une adresse peut y être affectée et donc elle sera perpétuellement accessible puisque cette interface est toujours active.
- Interface physique, l'adresse qui y est affectée n'est accessible qu'à la condition où l'interface soit activée, et qu'elle soit correctement branchée pour fournir une connectivité.

Si les interfaces sont connectées, et que des adresses IPv6 ont été affectées à ces interfaces en statique ou dynamiquement, alors elles apparaissent dans la table de routage, et de fait la machine dispose d'une connectivité vers les réseaux dont les préfixes ont étés affectés

- choix interface de sortie vers un voisin
  - A1--1B
  - 2 postes A et B sont connectés sur le même réseau en local,
  - l'attribution des adresses IPv6 avec le même préfixe les autorisent à échanger directement entre elles après le test d'adjacence

Test d'adjacence

Le test d'adjacence permet de vérifier si le destinataire partage un même réseau qui est accessible directement en utilisant les interfaces directement connectées:

- Pour cela la machine va comparer le préfixe de la destination avec les préfixes des réseaux directement connectés, si cela est vérifié la machine peut réaliser un routage direct. Le dispositif ICMPv6 découverte des voisins va permettre aux machines connectées sur le même réseau de se découvrir l'un et l'autre et de déterminer l'adresse physique d'un équipement à partir de son adresse IPv6. (Plus de détails sur cette fonction en Séquence n°3).
- Dans le cas présenté ci-dessous, les deux postes A et B, peuvent directement communiquer car ils sont connectés sur le même réseau à l'aide d'un commutateur, qui relaient de manière transparente les trames au niveau 2. Le préfixe IPv6 2001:db8:0001::/64 est paramétré sur chaque machine, donc les échanges sont possibles directement:

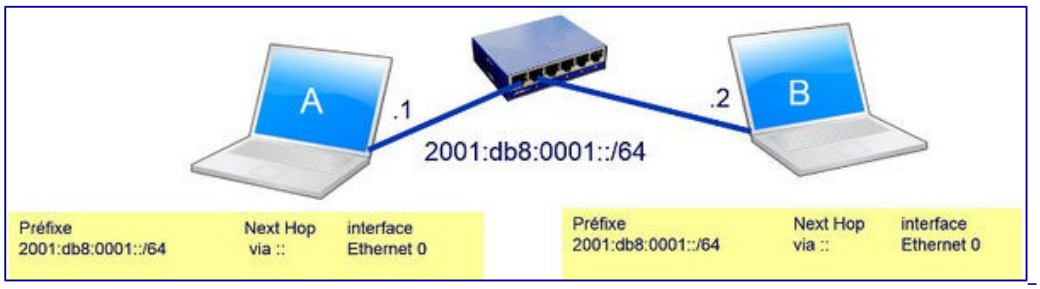

Figure 1: Routage statique direct

- Dans le cas contraire, un routage indirect s'impose, et la machine doit confier les paquets vers cette destination à une autre machine qui s'occupera de leurs acheminements, c'est le principe du routage indirect. Dans le cas présenté ci-dessous, le poste A peut atteindre les deux postes B et C, par contre B et C ne peuvent pas directement communiquer car ils sont connectés sur deux réseaux avec des préfixes IPv6 différents:

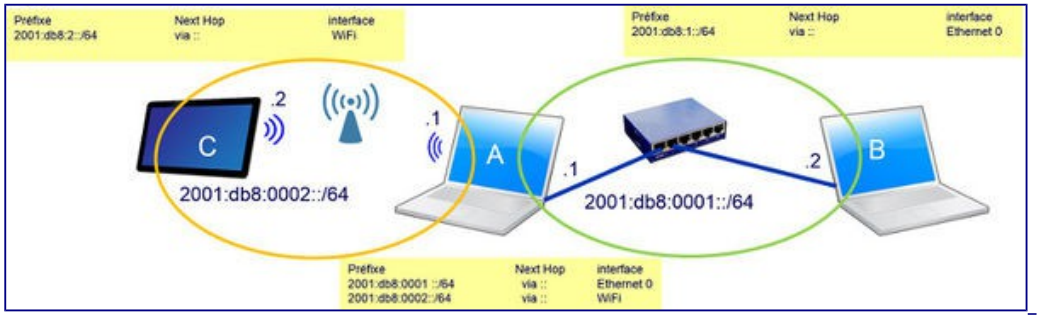

Figure 2: Routage statique indirect

- Donc dans la table de routage de B il faudra introduire une entrée vers le préfixe distant 2001:db8:0002::/64, en précisant l'adresse 2001:db8:0001::1/64 qui elle est directement accessible par B. Les paquets émis depuis B vers C seront dès lors retransmis.

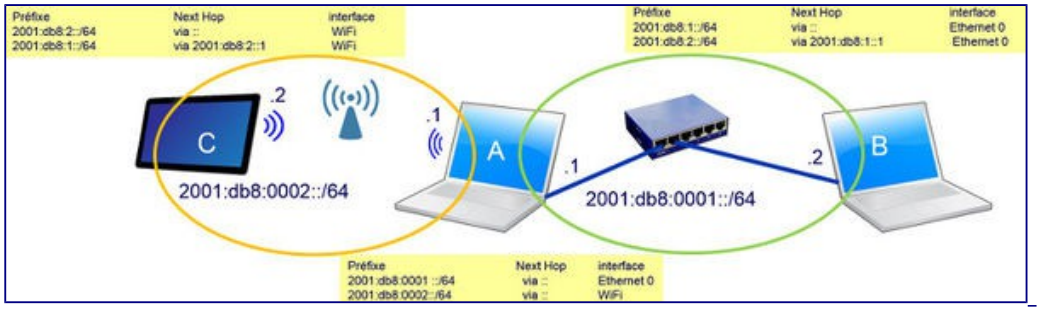

Figure 3: Routage statique indirect

- Ensuite il convient de ne pas omettre la même opération dans la table de routage de C, sans quoi aucune réponse vers B ne sera possible. Il faudra introduire une entrée vers le préfixe distant 2001:db8:0001::/64, en précisant l'adresse de A 2001:db8:0002::1/64 qui elle est directement accessible par C. Les paquets émis depuis C vers B seront dès lors retransmis par A.

Entrées d'une table de routage statique

La constitution d'une table de routage statique impose une configuration manuelle, afin de déterminer vers quelle passerelle l'équipement pourra se délester des paquets des destinations non directement connectées.

- Cas simple, une passerelle par défaut est spécifiée, et tous les paquets qui visent des destinations externes lui seront retransmis. En quelque sorte on fait confiance aux capacités et à la connectivité de cette passerelle. Une entrée de ce type est visible dans la table de routage de l'équipement, l'exemple suivant montre comment configurer une route statique par défaut sur un routeur IPv6:

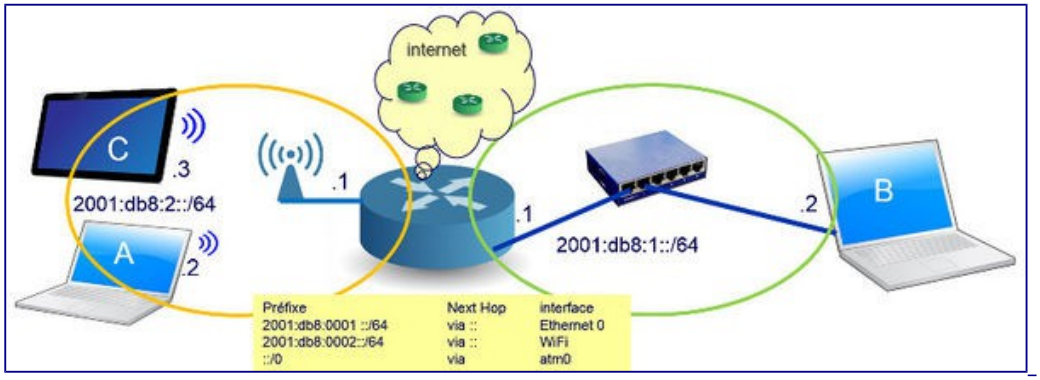

Figure 4: Routage par défaut

- Sur les postes de travail, il est donc simple de confier tous les paquets à destination de réseaux distants à la passerelle par défaut représentée par le routeur connecté à un fournisseur d'accès Internet. Une simple route par défaut est ajoutée à chaque poste de travail:

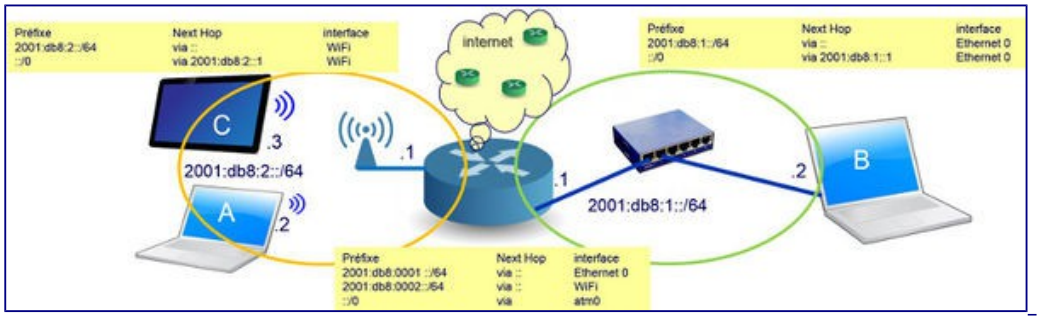

Figure 5: Routage par défaut

- Des routes spécifiques peuvent être définies, dès lors que l'on dispose d'une connectivité bien adaptée pour certains préfixes. Dance ce cas une configuration par des commandes de ce type sera nécessaire,

ipv6 route fd00:cafe:/48 fd00:2000::1

- Les routes les plus spécifiques c'est-à-dire celles avec un long préfixe seront traitées en premier, puis les routes moins spécifiques ensuite, et enfin la route par défaut en dernier ressort.

Routage dynamique

Comme dans IPv4, il faut faire la distinction entre deux grandes familles de protocoles de routage: les protocoles de routage internes (IGP: Interior Gateway Protocols) et externes (EGP: Exterior Gateway Protocols). C'est la notion de système autonome qui permet de faire la différence en définissant la portée des échanges d'informations de routage effectuée par ces protocoles de routage. Ainsi, la propagation des préfixes internes à un AS se fait par un IGP, alors que les annonces de préfixes entre AS se fait par un EGP.

Pour connecter un site à l'Internet, il faut donc mettre oeuvre des protocoles de routage internes et des protocoles de routage externes. Ce chapitre traite les trois protocoles IGP suivants: RIPng (équivalent de RIPv2 pour IPv4), ISIS et OSPFv3 (équivalent d'OSPFv2 pour IPv4), ainsi

que du protocole de routage externe BGP.

Les protocoles de routage internes permettent une configuration automatique des tables de routage des routeurs à l'intérieur d'un même système autonome. Les routeurs déterminent le plus court chemin pour atteindre un réseau distant. Les protocoles de routage internes nécessitent une configuration minimale du routeur notamment en ce qui concerne les annonces de routes initiées par ce routeur (ex. réseaux directement accessibles par une interface du routeur, annonces statiques ...).

Deux types de protocole de routage interne existent: les protocoles à état de lien ("link state" en anglais) et les protocoles à vecteur de distance ("distance vector" en anglais). Les premiers calculent le chemin le plus court en comptant le nombre de sauts pour atteindre le préfixe de destination, tandis que les seconds attribuent un coût à chaque lien en fonction de divers paramètres (type du lien...).

RIPng ou RIP IPv6

Les algorithmes appelés "distance vector", sont utilisés par le protocole de routage RIPv2 (RFC 2453). Ils sont basés sur l'algorithme de Bellman-Ford et figurent parmi les premiers algorithmes de routage interne utilisés dans l'Internet.

Les routeurs diffusent leurs tables de routage sur les liens auxquels ils sont connectés. Les autres routeurs modifient une route dans leur table si la métrique reçue (dans ce cas le nombre de routeurs à traverser pour atteindre une destination) est plus petite que celle déjà stockée dans la table. Si une annonce de route n'est pas présente dans la table, le routeur la rajoute. Ces modifications sont à leur tour diffusées sur les autres réseaux auxquels sont connectés les routeurs. Elles se propagent donc sur l'ensemble du réseau à l'intérieur du système autonome. On montre que cet algorithme converge et qu'en condition stable, aucune boucle n'est créée sur le réseau (c'est-à-dire qu'un paquet ne sera pas transmis indéfiniment de routeur en routeur sans jamais pouvoir atteindre sa destination).

Les tables sont émises périodiquement. Si un routeur tombe en panne ou si le lien est coupé, les autres routeurs ne recevant plus l'information suppriment l'entrée correspondante de leur table de routage. RIPng est le premier protocole de routage dynamique proposé pour IPv6 (RFC 2080) RIPng est une simple extension à IPv6 du protocole RIPv2 d'IPv4. Il en hérite les mêmes limitations d'utilisation (maximum de 15 sauts par exemple).

ISIS

IS-IS (Intermediate System to Intermediate System) est un protocole de routage interne à état de lien. Il a été standardisé par l'ISO (ISO 10589). C'est un protocole de niveau 3 (contrairement à OSPF et RIP qui sont de niveau 4) qui s'appuie sur une couche 2 de type Ethernet 802.2. Cet élément mérite d'être signalé car cela rend ce protocole indépendant d'IP, que ce soit IPv4 ou IPv6. Ce protocole travaille sur deux niveaux de hiérarchie: les aires (niveau 1) et le backbone (niveau 2).

Un routeur IS-IS peut être soit:

- level-1 (routage intra aire),
- level-2 (routage inter aire),
- level-1-2 (routage intra et inter aire).

Un routeur de niveau 1 n'a de voisins que dans son aire, alors qu'un routeur de niveau 2 peut avoir des voisins dans une autre aire. Il n'y a pas d'aire de backbone (contrairement à OSPF). Le backbone est constitué de la réunion de tous les routeurs de level-2. Sur un réseau de type LAN, il y a élection d'un routeur désigné (DIS) qui a la charge de produire les annonces.

Afin de construire sa topologie, IS-IS utilise 3 types de messages:

- les messages HELLO permettant de construire les adjacences;
- les messages LSP (Link State Protocol) permettant d'échanger les informations sur l'état des liens;
- les messages SNP (Sequence Number Packet) permettant de confirmer la topologie.

Pour élaborer ces messages, IS-IS se base sur l'utilisation d'éléments d'informations indépendants appelés TLV (Type, Longueur, Valeur). Le message est ainsi constitué d'un en-tête suivi d'une liste de TLV. Chaque TLV véhicule une information propre, et est donc standardisée. L'exemple ci-dessous montre une TLV Protocoles Supportés faisant partie d'un message HELLO, informant les voisins des protocoles supportés par l'émetteur du paquet:

- 0x81 0x02 0xcc 0x8e
    - Le premier octet donne le type de la TLV. Il s'agit ici du type 0x81, c'est-à-dire Protocoles supportés.
    - Le second octet donne la longueur en octets de la TLV: ici les deux octets qui suivent.
    - Les autres octets composent la valeur de la TLV. Ici nous avons deux octets indiquant des numéros de protocoles supportés (NLPID: Network Layer Protocol IDentifier): 0xCC pour IPv4 et 0x8E pour IPv6.

OSPFv3

Le troisième protocole de routage interne, basé sur l'algorithme du plus court chemin, s'appelle OSPF (Open Shortest Path First). Relativement plus difficile à mettre en oeuvre que RIPng, il est beaucoup plus efficace dans les détections et la suppression des boucles dans les phases transitoires. Ce protocole est basé sur plusieurs sous-protocoles, dont un qui permet une inondation fiable du réseau. Les routeurs possèdent alors chacun une copie des configurations de tous les routeurs présents sur le réseau, et peuvent calculer simultanément le plus court chemin pour aller vers l'ensemble des destinations.

Pour réduire la durée des calculs et surtout pour éviter un recalcul complet des routes à chaque changement de configuration, OSPF offre la possibilité de découper le réseau en aires. Une aire principale appelée backbone relie toutes les autres aires. Les réseaux trouvés dans une aire donnée sont envoyés aux autres aires par les routeurs qui sont en frontière d'aire.

OSPF a été adapté à IPv6 (RFC 2740); la version est passée de 2 à 3. La plupart des

algorithmes implementés dans OSPFv2 ont été réutilisés en OSPFv3; bien évidemment, certains changements ont été nécessaires en vue de l'adaptation aux fonctionnalités d'IPv6.

BGP

BGP4 est le protocole de routage externe actuellement utilisé pour le routage global de l'Internet IPv4 (la version 4, identique pour BGP et IP, est pure coïncidence). Compte tenu de sa criticité, ce protocole est l'objet d'évolutions constantes. L'une d'entre elles est le RFC 2858 qui rend BGP4 multi-protocole en introduisant la notion de famille d'adresse (ex. IPv4, IPv6, IPX...) et de sous-famille d'adresse (ex. unicast, multicast). Le RFC 2545 précise l'usage des extensions multi-protocoles pour le cas d'IPv6.

L'adaptation multi-protocole de BGP4 est assez simple car elle ne concerne que les trois attributs dont le format dépend de l'adresse soit:

- NLRI: Network Layer Reachability Informations (suite de préfixes);
- NEXT_HOP: Adresse IP où il faut router les NLRI;
- AGGREGATOR: Adresse IP du routeur qui a fait une aggrégation de préfixes.

Pour réaliser pratiquement cette adaptation, BGP4+ introduit deux nouveaux attributs:

- MP_REACH_NLRI: Multiprotocol Reachable NLRI;
- MP_UNREACH_NLRI: Multiprotocol Unreachable NLRI.

qui indiquent que l'on annonce des informations de routage autres que les routes unicasts IPv4. Ces attributs codent en premier le type de famille et de sous-famille d'adresse, puis les attributs dont le format est spécifique. Les autres attributs (comme le chemin d'AS Autonomous Systems) sont codés et annoncés sans changement.

Les implémentations du RFC 2858 sont souvent appelées MBGP (pour faire référence à leur capacité de traitement des routes multicast) ou BGP4+ (pour faire référence à leur capacité de traitement de routes IPv6). Pour l'anecdote, le numéro de version du protocole n'a pas été modifié (en BGP5 par exemple) car le passage de BGP3 à BGP4 rappelle trop de souvenirs douloureux à ceux qui l'ont mis en oeuvre. Les numéros d'AS utilisés pour IPv4 servent aussi pour IPv6.

Pour aller plus loin

Vous pouvez approfondir vos connaissances en consultant les liens suivants:

RIPng:

- http://livre.g6.asso.fr/index.php?title=RIPng
- RFC 2453 : RIP Version 2
- RFC 4822 : RIPv2 Cryptographic Authentication

ISIS:

- http://livre.g6.asso.fr/index.php?title=ISIS

- ISO-IEC 8473 Information technology — Protocol for providing the connectionless-mode network service: Protocol specification

OSPF:

- http://livre.g6.asso.fr/index.php?title=OSPFv3
- RFC 5340 : https://tools.ietf.org/html/rfc5340
- RFC 7503 : https://tools.ietf.org/html/rfc7503

BGP:

- http://livre.g6.asso.fr/index.php?title=BGP
- RFC 2545 : http://tools.ietf.org/html/rfc2545
- RFC 4760 : http://tools.ietf.org/html/rfc4760
- RFC 3849 : https://tools.ietf.org/html/rfc3849

Activité 24: Les mécanismes d'extensions IPv6

Introduction

Le besoin d'étendre l'en-tête IPv6 permet d'ajouter des fonctionnalités supplémentaires impliquant:

- Soit le destinataire du paquet IPv6
- Soit des routeurs intermédiaires étant impliqués dans l'acheminement de ce paquet IPv6

De nombreuses extensions ont été définies, nous allons proposer quelques exemples simples et démonstratifs de ce mécanisme.

Next Header

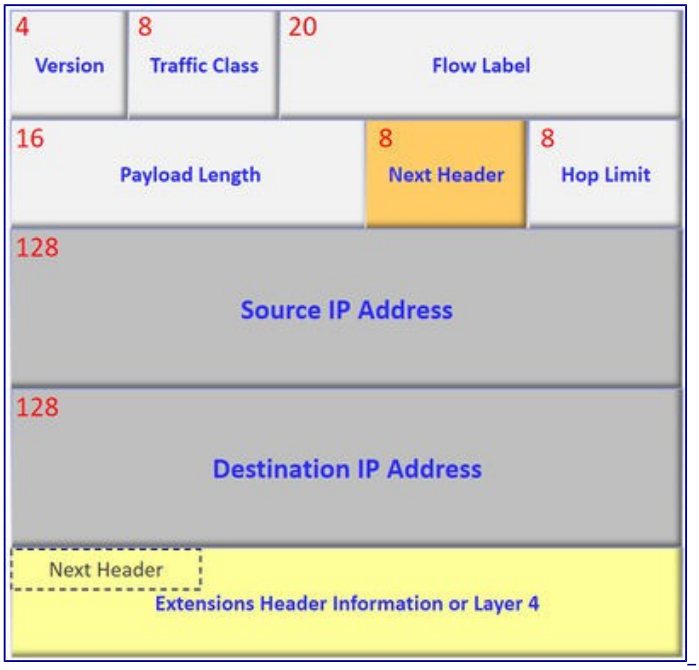

Figure 1: Localisation du champ Next Header dans l'en-tête IPv6.

Le champ Next Header désigne généralement les couches de protocoles de niveau supérieur comme par exemple le transport TCP/UDP, mais dans le cas des extensions plusieurs mécanismes particuliers sont également disponibles parmi la liste ci-après:

Next Header Value	Description
0	Proche en proche
4	IPv4
6	TCP
17	UDP
41	IPv6
43	Routage
44	Fragmentation
50	ESP confidentialité
51	AH Authentification
58	ICMPv6
59	IPv6 No Next
60	Destination
135	Mobilité
136	UDP-lite
140	Shim6
194	Jumbogramme

Tableau 1: Valeurs du champ Next Header.

La classification des extensions est fonction de la portée de celles-ci:

- Extension impliquant le destinataire: Destination, ESP, AH, Fragmentation
- Extensions impliquant tous les routeurs intermédiaires: Hop-by-Hop
- Extensions impliquant seulement certains routeurs désignés: Routing

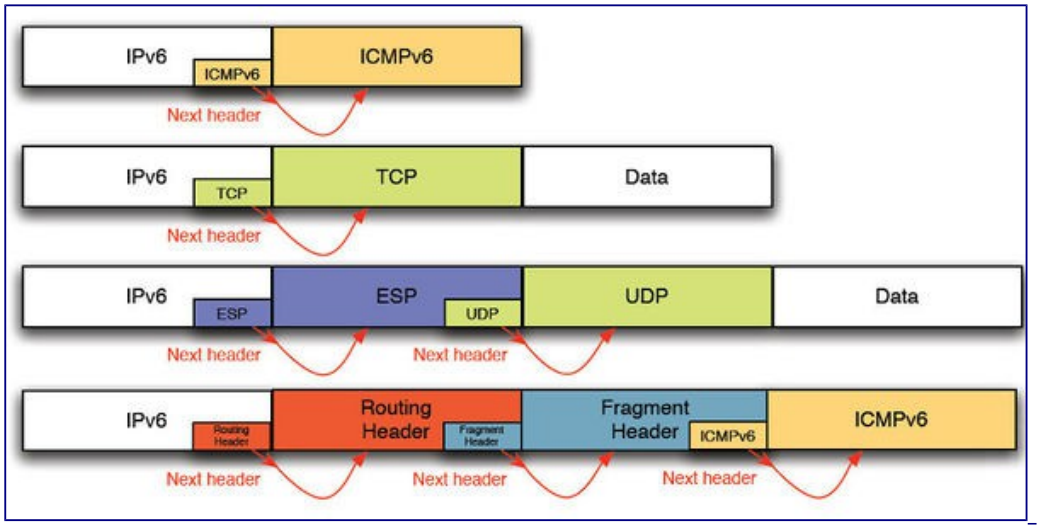

Figure 2: Enchaînement d'extensions

La figure 2 montre la souplesse avec laquelle plusieurs extensions peuvent être chaînées. Chaque extension contient dans son en-tête un champ en-tête suivant et longueur. Le premier paquet ne contient pas d'extension, le champ en-tête suivant pointe sur ICMPv6. Le second paquet ne contient pas d'extension, le champ en-tête suivant pointe sur TCP. Le troisième paquet contient une extension de protection qui pointe ensuite sur UDP. Dans le dernier paquet, une extension de routage, qui pointe sur une extension de fragmentation, qui finalement pointe sur ICMPv6.

- Si cet enchaînement d'extension offre beaucoup plus de souplesse que les options d'IPv4, il rend difficile la lecture des numéros de port, il faut en effet lire tout l'enchaînement d'extension pour arriver au protocole de niveau 4. Ceci a servi de justification à l'identificateur de flux qui permettait de refléter au niveau 3 un flux particulier et évitait de dérouler l'enchaînement. Bien entendu, les pare-feux devront vérifier les numéros de ports.
- Les extensions peuvent être vues comme un protocole 3.5 (entre la couche 3 et la couche 4). En effet, à part l'extension de proche-en-proche, qui est traitée par tous les routeurs traversés, les autres extensions ne sont traitées que par le destinataire du paquet (i.e. celui spécifié dans le champ adresse de destination du paquet IPv6).
- Si d'un point de vue théorique les extensions sont supérieures aux options d'IPv4, dans la réalité très peu sont utilisées à grande échelle ou restent du domaine de la recherche.

Quelques exemples

Extension de routage

Cette extension permet d'imposer à un paquet une route différente de celle offerte par les politiques de routage présentes sur le réseau. Pour l'instant seul le routage par la source (type = 0), similaire à l'option Loose Source Routing d'IPv4, est défini pour IPv6. La mobilité IPv6 a également introduit une autre extension de routage (type = 2; Optimisation dans le cas du mobile dans un réseau étranger).

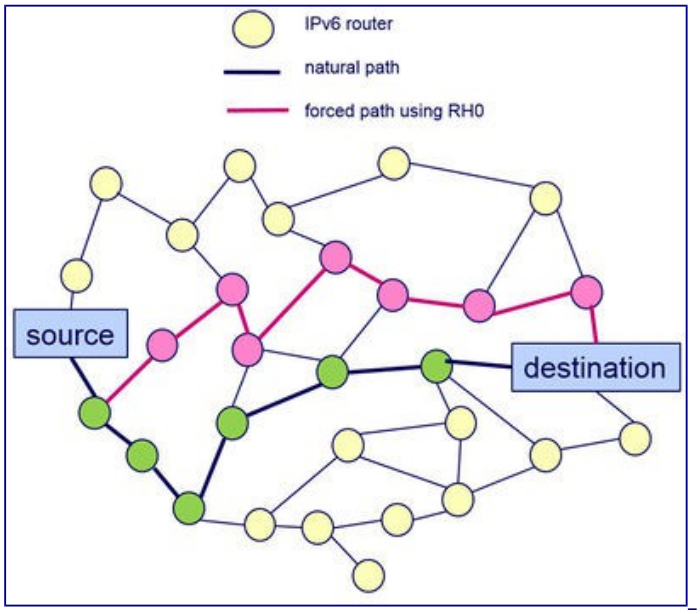

Figure 3: Enchaînement d'extensions.

Dans IPv4, le routage peut être strict (le routeur suivant présent dans la liste doit être un voisin directement accessible) ou libéral (loose) (un routeur peut utiliser les tables de routage pour joindre le routeur suivant servant de relais). Dans IPv6, seul le routage libéral est autorisé. En effet, le routage strict était initialement mis en place surtout pour des raisons de sécurité. La source devait être absolument sûre du chemin pris par les paquets. Cette utilisation a maintenant disparu du réseau.

Le principe du routage par la source ou Source Routing dans IPv4 qui vient d'être rappelé; est le même pour IPv6. L'émetteur met dans le champ destination du paquet IPv6, l'adresse du premier routeur servant de relais, l'extension contient la suite de la liste des autres routeurs relais et le destinataire. Quand un routeur reçoit un paquet qui lui est adressé comportant une extension de routage par la source, il permute son adresse avec l'adresse du prochain routeur et réémet le paquet vers cette adresse suivante.

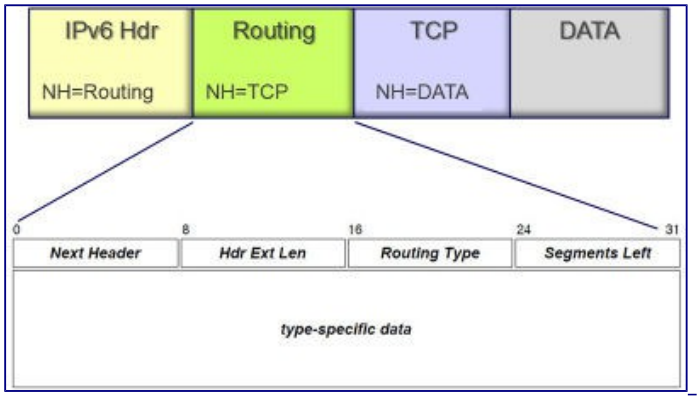

Figure 4: Extension de routage

La figure Format de l'extension routage par la source donne le format de l'extension de routage par la source:

- Le champ longueur de l'en-tête indique le nombre de mots de 64 bits qui composent l'extension. Pour l'extension de type 0, cela correspond au nombre d'adresses présentes dans la liste, multiplié par 2.
- Le champ type indique la nature du routage. Pour l'instant, seul le routage par la source, de type 0 est spécifié. La suite de l'en-tête correspond à ce type.
- Le nombre de segments restant est décrémenté après la traversée d'un routeur. Il indique le nombre d'équipements qui doivent encore être traversés. Il permet de trouver l'adresse qui devra être substituée.
- Les 32 bits suivants sont inutilisés pour préserver l'alignement.

La liste comprenant les routeurs à traverser et le destinataire est fournie. Ces adresses ne peuvent pas être multicast.

Dans l'exemple suivant, nous pouvons suivre l'évolution des changements des champs pendant la traversée du réseau du paquet IPv6:

- Noter l'évolution du champ Segment Left qui pointe vers l'adresse du prochain routeur spécifié apte à traiter l'extension RH0.
- Chaque routeur spécifié successif remplace l'adresse destination du datagramme avec l'adresse pointée par le champ Segment Left, une fois que le pointeur est décrémenté à 0, plus aucun changement ne sera effectué.
- Les routeurs non spécifiés relaient les paquets de manière transparente.

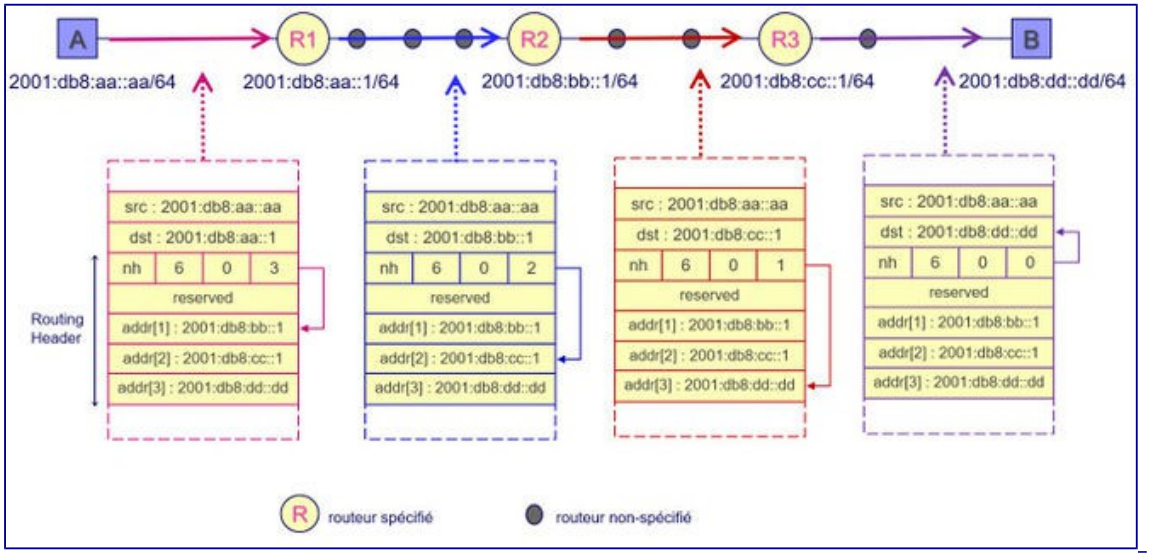

Figure 5: Extension de routage.

Fragmentation

La fragmentation telle qu'elle est pratiquée dans IPv4 n'est pas très performante. Initialement, elle servait à rendre transparente les limitations physiques des supports de transmission. Dans IPv4 quand un routeur ne peut pas transmettre un paquet à cause de sa trop grande taille et si le bit DF (don't fragment) est à 0, il découpe l'information à transmettre en fragments. Or le réseau IP étant un réseau à datagramme, il n'y a pas de possibilité de contrôler les fragments. Deux fragments successifs peuvent prendre deux chemins différents et par conséquent seul le destinataire peut effectuer le réassemblage. En conséquence, après la traversée d'un lien impliquant une fragmentation, le reste du réseau ne voit passer que des paquets de taille réduite.

Il est plus intéressant d'adapter la taille des paquets à l'émission. Ceci est fait en utilisant les techniques de découverte du MTU (voir RFC 1981 : Mécanisme de découverte du PMTU). En pratique une taille de paquets de 1 500 octets est presque universelle.

Il existe pourtant des cas où la fragmentation est nécessaire. Ainsi une application telle que NFS sur UDP suppose que la fragmentation existe et produit des messages de grande taille. Comme on ne veut pas modifier ces applications, la couche réseau d'IPv6 doit aussi être capable de gérer la fragmentation. Pour réduire le travail des routeurs intermédiaires, la fragmentation se fera chez l'émetteur et le réassemblage chez le récepteur.

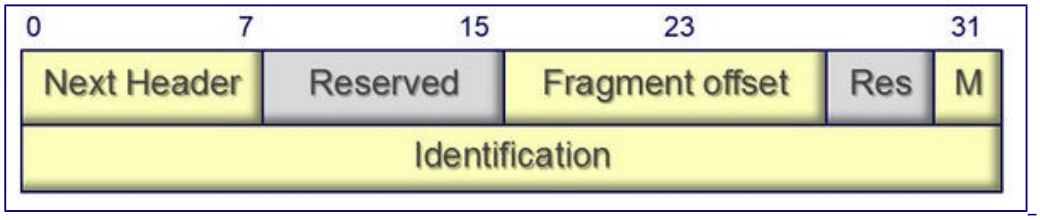

Figure 6: Format de l'extension de fragmentation

Authentification

L'extension ESP (Encapsulating Security Payload) décrite dans le RFC 2406 permet de chiffrer

l'ensemble des paquets ou leur partie transport et de garantir l'authentification et l'intégrité de ces paquets. Cette extension permet optionnellement de détecter les rejeux (à condition que le service d'authentification soit assuré) et garantit de façon limitée la confidentialité du flux.

Pour obtenir ces services de sécurité, il est nécessaire, avant d'émettre un paquet IP sur le réseau, de chiffrer les données à protéger, de calculer un authentificateur et d'encapsuler ces informations dans l'en-tête de confidentialité. Cela nécessite bien entendu l'existence d'une association de sécurité précisant entre autres le(s) algorithme(s) de chiffrement, la (les) clé(s) et un indice de paramètres de sécurité. L'extension ESP est composée d'un en-tête ESP, d'une queue ESP et d'un authentificateur ESP. Suivant le mode de protection sélectionné, l'étendue des champs protégés diffère:

- En mode transport, seules les données de niveau transport du paquet IP (de type TCP, UDP, ICMP) sont protégées. Plus précisément, le chiffrement porte sur les données et sur la queue ESP avec la possibilité de porter sur l'extension destination à condition que celle-ci soit placée dans l'extension ESP (cf. figure Positionnement de l'extension ESP en modes transport et tunnel). La protection en intégrité/authentification porte sur toute l'extension ESP excepté l'authentificateur placé dans le champ authentificateur. Elle assure ainsi la protection des données de niveau transport du paquet IP. L'extension ESP est insérée dans le paquet IP juste après l'en-tête du paquet IP et les extensions avec la possibilité d'avoir l'extension destination placée juste avant l'extension ESP ou encapsulée dans l'extension ESP en première position.
- En mode tunnel, la protection porte sur tout le paquet IP original. C'est-à-dire, le paquet IP original est chiffré avant d'être encapsulé dans l'extension ESP. Cette extension est alors placée dans un nouveau paquet IP dont les en-têtes IP sont en clair (non chiffrés) pour permettre au réseau IP de correctement acheminer le paquet (cf. figure Positionnement de l'extension ESP en modes transport et tunnel). Il s'agit du classique chiffrement des artères. La protection en intégrité/authentification porte sur l'extension ESP (comprenant le paquet IP original) excepté l'authentificateur ESP. L'extension ESP rend le service de confidentialité du flux de façon limitée en ce sens que ce service n'est rendu qu'en mode tunnel et ne porte que sur la confidentialité des adresses IP. En effet, l'extension ESP en mode tunnel réalise le chiffrement de l'intégralité des paquets IP originaux, y compris leurs adresses IP. De ce fait, si les équipements qui mettent en oeuvre IPsec ne sont pas l'émetteur et le destinataire final des paquets, leurs adresses masqueront celles de ces derniers. Un intrus placé en écoute sur le réseau au niveau d'un tunnel IP (entre les passerelles de sécurité) ne pourra pas dans ce cas-là connaître les adresses des stations en communication.

Pour aller plus loin

Vous pouvez approfondir vos connaissances en consultant les liens suivants:

- RFC 2460 : http://tools.ietf.org/html/rfc2460#section-4.1
- RFC 2095 : http://tools.ietf.org/html/rfc5095

Activité 25: La taille des paquets IPv6

Introduction

La couche réseau a pour tâche de placer les segments provenant de la couche transport (données utiles + en-tête transport) dans des paquets. Ces paquets sont ensuite placés dans des trames avant d'être émis sur le support physique. La gestion de la taille du paquet sur le chemin est nécessaire dès lors que la communication emprunte des liaisons de nature différente tout au long du parcours.

Cas nominal (taille paquet PMTU)

La couche réseau a pour tâche de placer les segments provenant de la couche transport (données utiles + entête transport) dans des paquets. Ces paquets sont ensuite placés dans des trames sur le support physique. Ce support, selon sa nature, définit une taille maximale de trame: Ethernet (1500 octets), PPPoA (1468 octets), MPLS (de 1500 à 65535 octets), etc. Cette taille fixe donc pour la couche réseau la taille maximale de données pouvant être placées dans un paquet, appelée MTU (*Maximum Transmission Unit*).

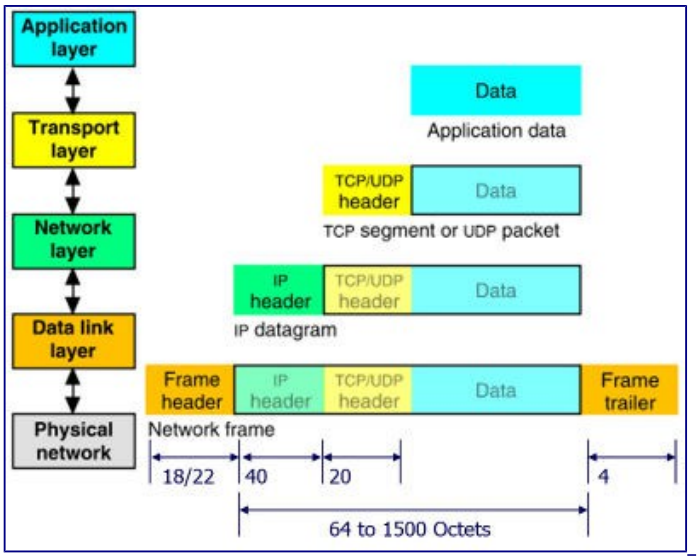

Figure 1: Encapsulation IP dans Ethernet.

Un paquet IP est cependant amené à voyager sur plusieurs supports de natures différentes, chacun imposant des tailles maximales différentes. Pour pouvoir parcourir son chemin jusqu'à sa destination, le paquet doit donc avoir une taille inférieure ou égale à la plus grande taille autorisée par l'ensemble des liens traversés. Cette taille est de ce fait appelée PMTU (Path Maximum Transmission Unit) ou unité de transfert de taille maximale sur le chemin.

Pour des considérations d'efficacité, il est généralement préférable que les informations échangées entre équipements soient contenues dans des datagrammes de taille maximale. Une trop petite taille de paquet a pour effet d'augmenter la charge supplémentaire des en-têtes par rapport aux données transportées, ainsi que d'augmenter le nombre de paquet à traiter dans les routeurs. Au moment de créer des paquets, la couche réseau essaie donc de respecter au maximum la PMTU.

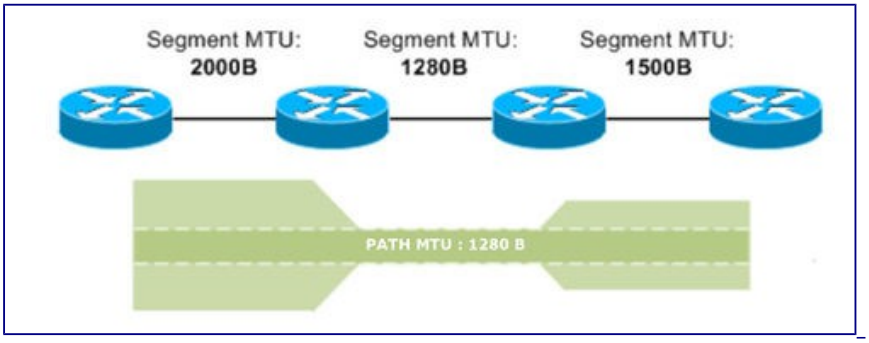

Figure 2: Path MTU

Il est à noter qu'IPv6 impose une valeur minimale pour la MTU au niveau réseau (et donc pour la PMTU pour un chemin), valeur fixée à 1280. Cette limite a pour objectif d'éviter qu'un lien imposant une MTU très faible n'implique la transmission de petits paquets pour tous les chemins empruntant ce lien. Si un support physique impose une taille inférieure à 1280, il est nécessaire de mettre en place une couche d'adaptation pour la couche réseau. C'est le cas par exemple pour les réseaux IEEE 802.15.4 (imposant une MTU de 81 octets), pour laquelle la couche d'adaptation pour IPv6 6LowPAN (RFC 4944) a dû être spécifiée.

Cas où taille paquet PMTU

Cependant la valeur de la PMTU n'est pas forcément connue à l'envoi d'un premier paquet vers une destination quelconque. L'émetteur du paquet fait alors la supposition que la taille maximale vers cette destination est égale à celle du support physique sur lequel il est connecté, c'est à dite la MTU du réseau d'accès.

L'acheminement du paquet s'effectue normalement jusqu'au premier routeur rencontrant une incompatibilité entre la taille du paquet à transmettre et la taille maximale autorisée sur le support physique. Le routeur est alors dans l'incapacité de transmettre le paquet. Dans le cas d'un paquet IPv6, le routeur utilise alors un message de signalisation (basé sur ICMPv6, qui sera décrit dans la séquence 3) pour informer l'émetteur du paquet du problème d'acheminement, ainsi que de la taille recommandée pour que les paquets soient retransmis.

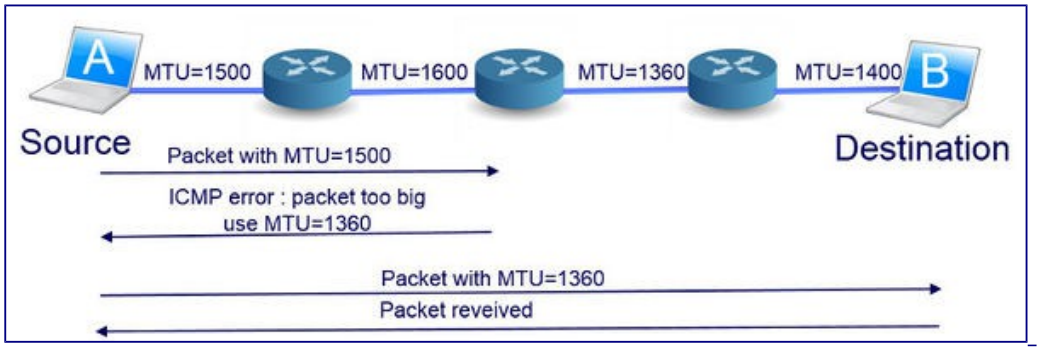

Figure 3: Négociation Path MTU Discovery.

La couche réseau de l'émetteur doit alors, à la réception de ce message, émettre de nouveau les données, mais dans un paquet ayant pour taille celle recommandée dans le message. Cette valeur est alors enregistrée comme la PMTU pour tous les prochains paquets vers cette même destination. Ce processus peut être répété si d'autres liens imposent des tailles maximales de

transmission encore inférieures. L'émetteur enregistrera les valeurs recommandées successives jusqu'à arriver à la taille maximale autorisée sur le chemin. Les paquets suivants qui respecteront cette taille seront alors acheminés sans problème. Ce mécanisme de découverte de la taille maximale de transmission sur le chemin est spécifié dans le RFC 1981 .

Besoin de fragmentation IPv6

Il existe cependant des cas où la couche réseau ne peut pas adapter la taille des données à transmettre à la taille maximale autorisée sur le chemin. C'est le cas par exemple des messages transportés sur UDP pour le système de fichier NFS. Ces messages peuvent avoir une taille supérieure à celle autorisée sur le support.

La couche réseau n'a alors d'autres choix que de fragmenter ces données. Le principe de la fragmentation est de séparer un paquet devant être émis avec une taille trop importante en plusieurs paquets respectant la taille maximale autorisée. Ces paquets (ou fragments) sont émis et acheminés vers la destination comme n'importe quel autre paquet IP. La couche réseau du destinataire se charge alors de reconstruire le paquet IP original pour que les données puissent être traitées

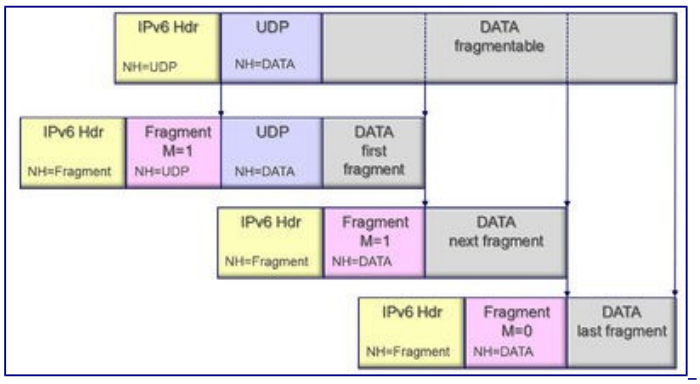

Figure 4: Fragmentation.

L'identification d'un fragment (à quel paquet appartient-il? Quel est la position relative de ce fragment?) est transmise dans une extension de fragmentation de l'en-tête IPv6. Le format de l'extension de fragmentation est donné figure Format de l'extension de fragmentation.

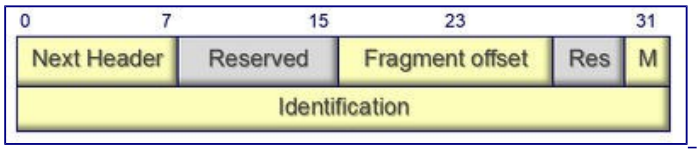

Figure 5: Format de l'extension de fragmentation

- Le champ `place du fragment` indique lors du réassemblage où les données doivent être insérées. Ceci permet de parer les problèmes dus au déséquencement dans les réseaux orientés datagrammes. Comme ce champ est sur 13 bits, la taille de tous les segments, sauf du dernier, doit être multiple de 8 octets.
- Le bit `M` s'il vaut 1 indique qu'il y aura d'autres fragments émis.
- Le champ `identification` permet de repérer les fragments appartenant à un même paquet initial. Il est différent pour chaque paquet et recopié dans ses fragments.

Note: la fragmentation est permise en IPv4 au niveau des routeurs intermédiaires, leur donnant ainsi la possibilité de transmettre un paquet même s'il est de taille supérieure à la MTU du lien suivant. Mais dans ce cas, le mécanisme est jugé inefficace, car il augmente la tâche des routeurs. En IPv6, grâce au mécanisme de découverte de la taille maximale de transmission, les routeurs intermédiaires ne fragmentent plus les paquets. Si la fragmentation est cependant nécessaire, cette tâche est déléguée aux extrémités de la communication.

Jumbogrammes

Une autre fonction optionnelle d'IPv6, est l'option jumbogramme dans une extension d'en-tête Hop-By-Hop, qui permet l'échange de paquets ayant une charge utile jusqu'à 4 GB moins un (2^32 − 1 = 4294967295 octets), en permettant l'utilisation d'un champ longueur de 32-bit. De tels paquets sont appelés jumbogrammes.

Étant donné que TCP et UDP disposent de champs limités à 16 bits (longueur, pointeur urgent), le support des jumbogrammes IPv6 nécessite des modifications sur l'implémentation des couches de protocoles Transport. Les jumbogrammes sont intéressants sur des liens qui disposent d'un MTU plus grand que 65583 octets (plus de 65535 octets de charge utile, plus 40 octets pour la taille fixe de l'en-tête, plus 8 octets pour l'en-tête d'extension Hop-by-Hop).

Pour aller plus loin

Vous pouvez approfondir vos connaissances en consultant les liens suivants:

- RFC 1981 : Path MTU Discovery for IP version 6
- RFC 2460 : https://tools.ietf.org/html/rfc2460#section-4.5
- RFC 4944 : Transmission of IPv6 Packets over IEEE 802.15.4 Networks

Conclusion

Grâce à cette deuxième séquence du Mooc IPv6 vous avez découvert et appréhendé différents aspects du protocole:

- Après avoir passé en revue le format de l’en-tête des paquets IPv6,
- Vous avez compris l'importance des mécanismes d’encapsulation,
- Vous avez intégré les principes de routage,
- Vous avez appréhendé les extensions de l’en-tête IPv6
- Enfin après avoir mis en oeuvre une configuration simplifiée

Dorénavent vous êtes aptes à approfondir d'autres mécanismes importants pour faciliter l'intégration du protocole dans toutes les infrastructures où IPv6 sera utile d'être déployer. C'est bien ce que vous allez découvrir dans les prochaines séquences.