UFR sciences et techniques
2, rue de la Houssinière
BP 92208
44322 Nantes Cedex 3

Stage du 24 janvier 2005 au 30 mai 2005 en alternance (deux matinées par semaine)

Promotion : 2004 - 2005

Filière : Master 1 informatique

Tuteur : Attiobgé Christian

Segmentation des flux d'un réseau local (VLAN) avec mise en place de liens d'agrégations et implémentation du *spanning tree*

ADJIDO Idjiwa
GIRAUD Laurent

CEntre de Recherche Méthodologique d'Architecture (CERMA)
UMR CNRS 1563
EAN - Rue Massenet
BP 81931
44319 Nantes cedex 3, France
Tuteur entreprise : Thomas LEDUC, Frédéric REINOLD

« *L'impensable est fait, l'urgent est en cours ;*
Pour l'impossible, merci de prévoir un petit délai... »

1. Remerciements

Traditionnellement, nous tenons à remercier ceux qui nous ont permis de réaliser ce stage dans de bonnes conditions.

Ainsi, nos remerciements vont à l'équipe du CERMA qui nous a accueilli avec sourires et disponibilité, même lorsque nous interrompions momentanément le bon fonctionnement du réseau. ☺

Un remerciement particulier à M. Leduc qui avait parfois du mal à se dédoubler.

Un tout aussi grand remerciement à M. Attiogbé qui nous a également toujours trouvé une place parmi tous les autres stagiaires afin de nous prodiguer conseils et corrections.

Merci à Mme Hamma pour les liens généreusement fournis sur lacp.

Enfin, merci à Caroline, Eric, Franck, Marc et Mehdi (les stagiaires qui ont partagés nos locaux) pour leur patience et compréhension lorsque M. Leduc n'était pas disponible pour eux et pour avoir contribué à la bonne ambiance des séances au CERMA.

Le binôme.

Sommaire

2. Introduction

Ce stage rentre dans le cadre de la formation de Master d'Informatique 1ère année de la faculté des sciences de Nantes. Un module du second semestre consiste en un stage ou en un travail d'étude et de recherche (TER). Désirant faire un Master 2 professionnel, nous avons opté pour la solution stage qui nous a paru plus « pratique » et d'un apport plus intéressant pour notre expérience professionnelle. Le sujet du CERMA (centre de recherche méthodologique d'architecture) situé sur le campus de l'école d'architecture, nous a paru correspondre à nos attentes. L'objectif est principalement de mettre en place des VLANs (réseaux locaux virtuels) sur quatre commutateurs *hp procurve* en installant une agrégation de liens entre les différents commutateurs. Le réseau étant en exploitation, utilisé par le personnel du CERMA, nous avons eu comme première contrainte d'opérer sur le matériel en dérangeant le moins possible le CERMA durant les deux matinées où nous sommes présents par semaine sur le site. En entreprise, nous avons été encadrés par M. Reinold Frédéric (doctorant) et M. Leduc Thomas (administrateur réseaux). A la faculté, c'est M. Attiogbé Christian (enseignant-chercheur) qui s'est chargé de nous aiguiller.

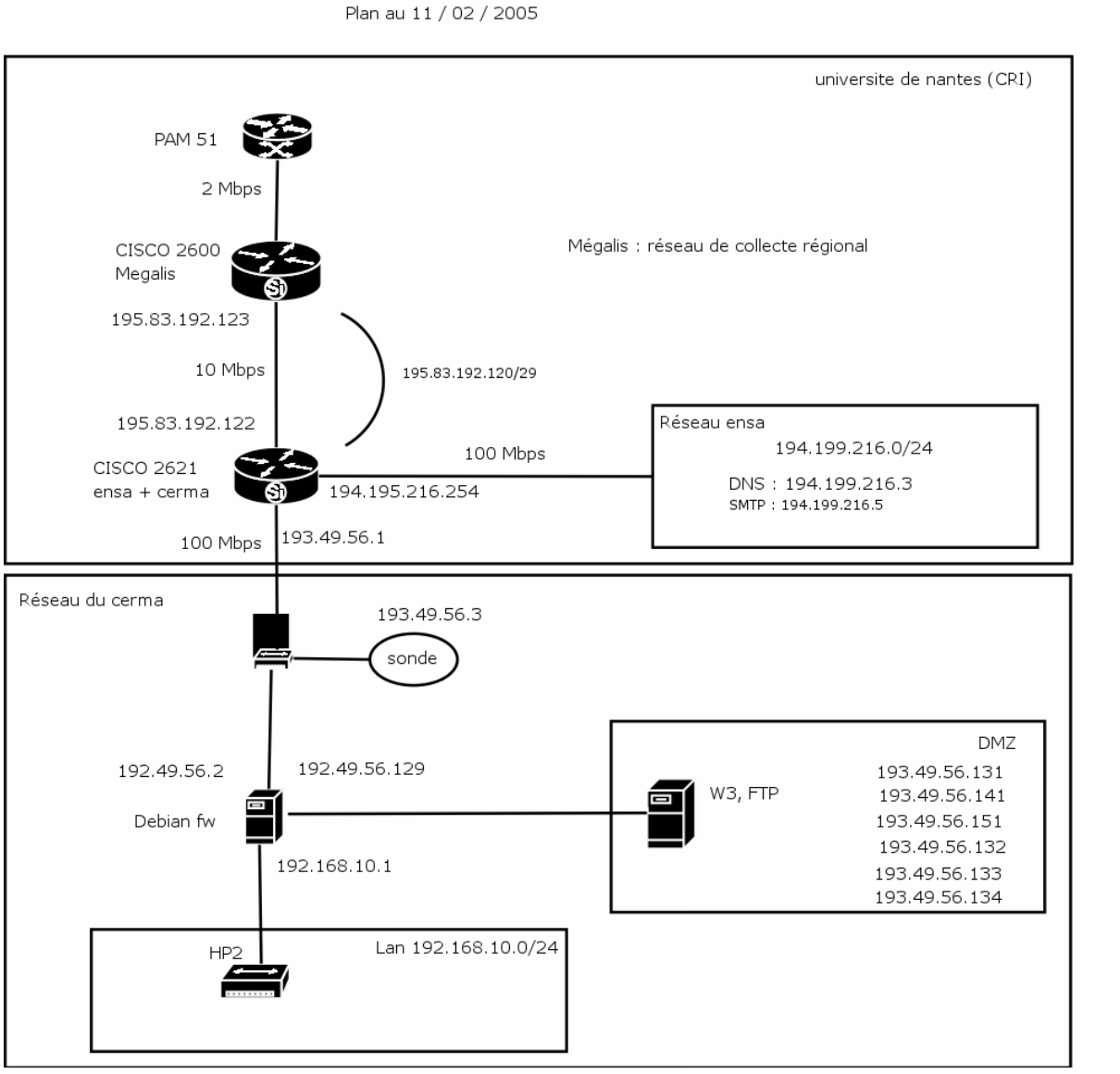
Le présent rapport expose les différentes étapes qui nous ont permis de répondre au cahier des charges. Nous commencerons par une rapide présentation du CERMA (du réseau en particulier) afin de préciser le contexte ; nous présenterons ensuite dans une première partie l'agrégation de liens en rappelant son utilité puis son implémentation entre les quatre commutateurs *hp procurve* de différentes générations (2524 et 2626). Dans une seconde partie nous présentons les réseaux locaux virtuels et leur implantation au CERMA. Nous finirons par montrer en quoi l'algorithme du *spanning tree* nous a été nécessaire et comment nous l'avons implémenté en fonction des moyens que nous avions.

3. Contexte du stage

Il s'agit ici de rappeler brièvement le contexte puis de présenter l'état du réseau à notre arrivée avant d'exposer ce qui a constitué principalement notre cahier des charges.

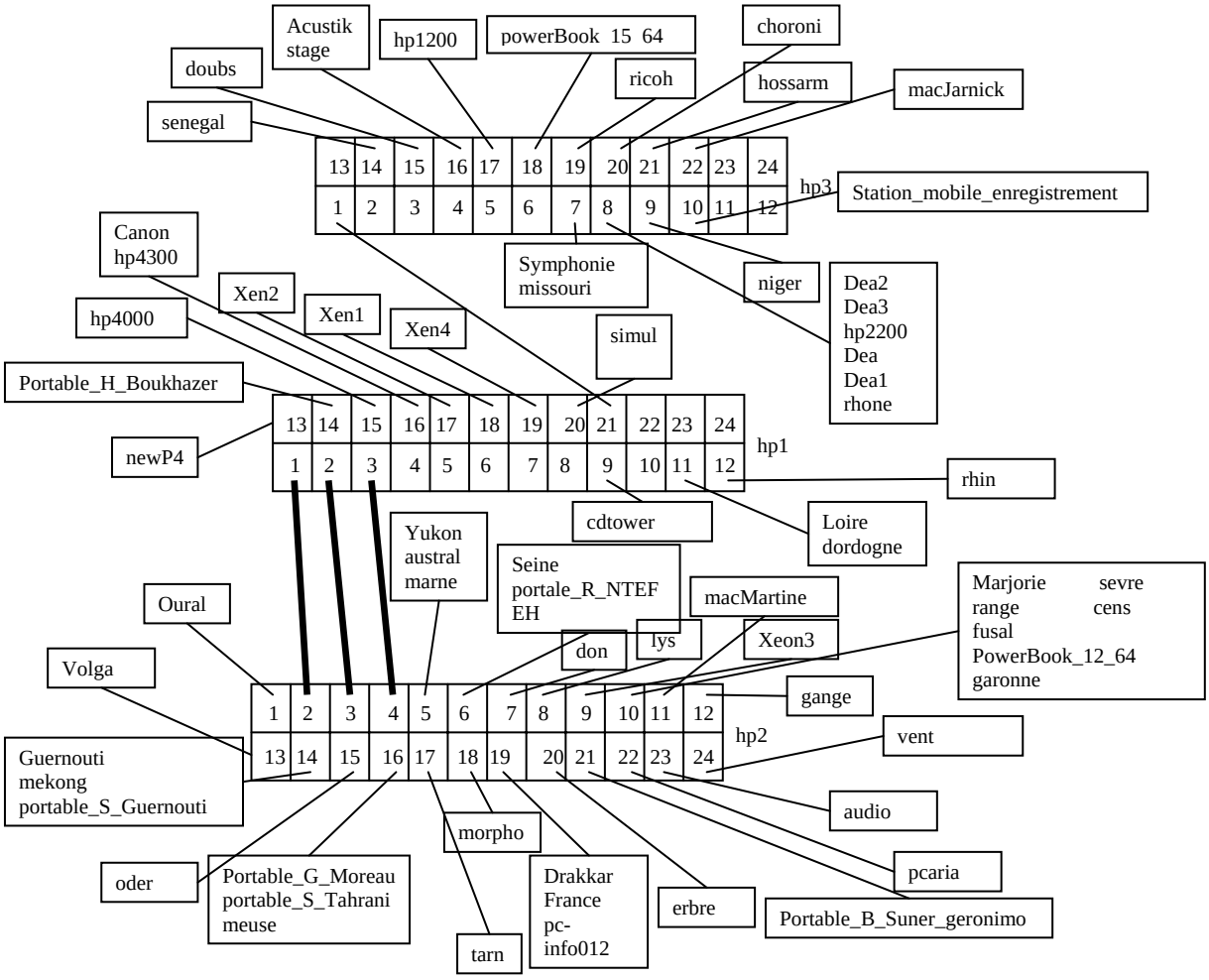
3.1. Le CERMA

Le centre de recherche méthodologique d'architecture est intégré à l'école d'architecture de Nantes. Le parc informatique du laboratoire est dissocié de celui de l'école. Le CERMA regroupe une cinquantaine de machines, qui sont cependant reliées à l'école d'architecture. Nous nous sommes préoccupés uniquement du réseau local au CERMA (lan 192.168.10.0/24).



3.2. Le réseau local du CERMA

Le réseau sur lequel nous avons commencé à travailler est le suivant :



Nous pouvons constater qu'une agrégation sur trois liens est en place entre hp1 et hp2 via les ports 1-3 (de 1 à 3) de hp1 et 2-4 de hp2.

3.3. Notre mission

Nous l'avons acceptée et effectuée avec enthousiasme. Elle consiste dans un premier temps à identifier le câblage existant, les postes, les éléments actifs et à se familiariser avec le réseau. Puis, munis d'un quatrième commutateur plus performant que les autres (un *hp procurve* 2626) nous devons établir des liens d'agrégation entre ces éléments actifs. Ensuite, après avoir rajouté des liens supplémentaires entres les commutateurs pour redondance en cas de ruptures des liens agrégés, nous devrons implémenter l'algorithme du *spanning tree*. Enfin nous structurerons le réseau en VLANs afin de séparer notamment les flux d'impression et d'administration du trafic utilisateur. Les différents besoins (listés dans le cahier des charges fournis en annexe) peuvent être récapitulés ainsi :

- B1 : Etiquetage du câblage ;
- B2 : Mise en place d'une agrégation LACP[1] entre les commutateurs ;
- B3 : Mise en place de l'algorithme du *spanning tree* ;
- B4 : Création et migration vers trois VLANs : administration, utilisateurs, imprimantes.

4. Identification et étiquetage du câblage

4.1. Identification des machines

La première étape a été de repérer le câblage du réseau : nous avons alors observé que l'étiquetage qui était en place ne correspondait plus à la réalité. Afin de pouvoir identifier plus facilement les machines, nous avons défini une nouvelle nomenclature d'étiquetage puis reétiqueter tout le réseau. Pour nous permettre de réaliser ce qui nous était demandé, nous avons dû installer du câblage supplémentaire entre les commutateurs.

4.2. Nomenclature

Notre nomenclature pour les étiquettes est basée sur le format suivant : **XY**
Où X correspond au numéro du commutateur (par exemple 1 pour le commutateur Hp1, 2 pour le commutateur Hp2, ...) et Y correspond au numéro du port auquel est connecté le câble. Ainsi, le câble relié au port 21 de Hp1 aura comme étiquette 121. Nous avons également défini un étiquetage spécifique pour les liaisons inter commutateurs. Ces câbles sont étiquetés de la manière suivante : **X Y – X' Y'**
Avec X, Y et X', Y' respectant la nomenclature définie ci-dessus. Nous avons donc pour la liaison du port 23 de Hp1 vers le port 6 de Hp2, une étiquette sur laquelle on peut lire 123-206. Ces étiquettes sont imprimées en rouge pour encore mieux différencier les liaisons inter commutateurs. Nous noterons que la nomenclature est spécifique au CERMA qui ne possèdent que quatre commutateurs et ne devrait pas faire l'acquisition de plus de cinq commutateurs supplémentaire. Nous sommes donc à l'abri d'une ambiguïté qui pourrait exister sur le 121 : on peut en effet s'interroger sur le fait qu'on puisse lire 1 et 21 ou 12 et 1 ce qui n'aurait plus la même signification.

[1] *Link Aggregation Control Protocol*

5. Agrégation de liens

5.1. Qu'est ce que l'agrégation de liens

5.1.1. Une norme

L'agrégation de liens est définie dans la norme IEEE 802.3ad ; elle permet d'augmenter la bande passante disponible entre deux stations Ethernet en autorisant l'utilisation de plusieurs liens physiques comme un lien logique unique. Ces liens peuvent exister entre 2 commutateurs ou entre un commutateur et une station. Avant cette norme, il était impossible d'avoir plusieurs liens Ethernet sur une même station, sauf si ces liens étaient reliés à des réseaux ou des VLANs différents.

L'agrégation de liens (appelé également *link aggregation* ou *port trunking*) apporte les avantages suivants :

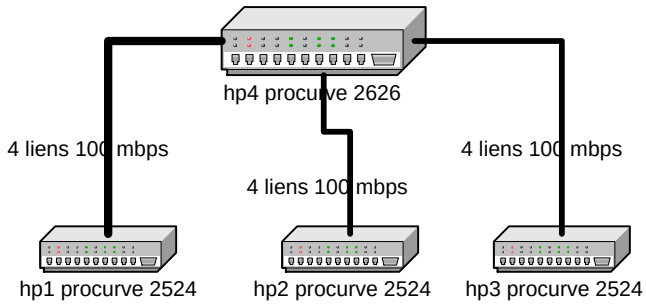
- La bande passante peut être augmentée à volonté, par pallier. Par exemple, des liens Fast Ethernet additionnels peuvent augmenter une bande passante entre deux stations sans obliger le réseau à passer à la technologie Gigabit pour évoluer ;
- La fonction de « load balancing » (équilibrage de charge) peut permettre de distribuer le trafic entre les différents liens ou au contraire de dédier une partie de ces liens (et donc de la bande passante) à un trafic particulier ;
- La redondance est assurée automatiquement : le trafic sur une liaison coupée est redirigé automatiquement sur un autre lien.

5.1.2. La répartition du trafic

Les interfaces Ethernet sont considérées comme une interface unique (une seule adresse MAC) avec tous ces liens physiques, la fonction d'agrégation de liens est donc totalement transparente pour les couches de haut niveau comme pour les protocoles de routage. Par contre, pour conserver l'ordre d'arrivée des trames à leur destinataire, les algorithmes chargés de l'agrégation créent des sessions appelées « conversations » qui regroupent les trames Ethernet ayant les mêmes adresses sources et destinations (couple **S**ender **A**ddress/**D**estination **A**ddress). Les trames d'une même conversation sont alors limitées à un seul lien physique. En d'autres termes, les données d'une même adresse source vers une même adresse destination se feront à travers le même lien. L'émission vers une autre destination se fera via un autre lien. Il est donc possible qu'un seul des liens soit utilisé complètement et que les autres ne le soient pas du tout. Nous pouvons également préciser que tous les liens physiques d'un même groupe doivent opérer en point à point, entre deux stations full-duplex et que tous les liens doivent fonctionner avec le même débit.

5.2. *Scénario pour le* CERMA

Le CERMA souhaite mettre en place une agrégation de liens entre les quatre commutateurs. Elle permettra comme indiqué plus haut d'augmenter la bande passante en même temps qu'elle mettra en place une redondance entre les liens reliant les commutateurs. Seul le commutateur 2626 (intitulé hp4) possède deux « ports gigabits » (i.e. assurant un débit de 1 Gigabit). Aucune agrégation n'est donc possible via des liaisons gigabits entre deux commutateurs du CERMA. De plus, les trois commutateurs 2525 (hp1, hp2 et hp3) ne peuvent implémenter qu'une seule agrégation à la fois. En revanche, hp4 peut traiter jusqu'à six agrégations. Les agrégations se feront de hp1, hp2, hp3 vers hp4 sur des ports à 100 Mbps. Quatre liens constitueront le lien logique, nous aurons donc des débits théoriques pour un couple d'adresse émission/destination de 400 Mbps. Soit 800 Mbps entre deux stations puisque nous sommes en full duplex. Le chiffre 4 est dû à une limite matérielle imposée par les commutateurs pour le nombre de ports par regroupement de port.
Le scénario à réaliser peut se présenter comme suit :



5.3. *Implémentation sur le matériel du* CERMA

La manipulation en détails et les différentes étapes (*step by step*) pour mettre en place l'agrégation de liens sont présentées en annexe (mode opératoire de l'implémentation de l'agrégation). Nous donnons ici les principales étapes avec les lignes de commandes utiles pour la compréhension du concept.

5.3.1. Sélection des ports concernés par l'agrégation

Les ports utilisés pour le regroupement (*trunk*) ne sont pas forcément consécutifs. Nous rappelons qu'il est essentiel que les ports en question soient dans le même mode (même vitesse, full duplex). Tout comme nous pouvons souligner que tout port rajouté à un *trunk* perd ses configurations de « *port security* ». Les ports sélectionnés pour l'agrégation sont les ports 1-4 (de 1 à 4) sur hp1 et hp3, 2-5 sur hp2, le port 1 étant pris pour la connexion à Oural (la passerelle de sortie vers le réseau Internet). Les ports 1-12 sont pris sur hp4, 1-4 pour l'agrégation avec hp1, 5-8 pour l'agrégation hp2 et 9-12 pour l'agrégation hp3.

5.3.2. Configuration du port trunking

La première étape a donc consisté à mettre en place les *trunks* de façon statique, sans utiliser de protocole particulier. Ce qui se fait comme suit :

```
HPswitch(config)# trunk <liste des ports> <trk1 | ... | trk6>  trunk
```

Le paramètre `<liste des ports>` est l'ensemble des ports regroupés pour former le *trunk*. Viens ensuite l'identifiant du *trunk* (jusqu'à six *trunks* possible pour hp4, un seul pour hp1, 2 et 3) et enfin le mot clé `trunk` qui indique qu'il s'agit d'un *trunk* statique sans protocole d'agrégation particulier. En effet il permet uniquement d'obtenir une redondance de liens (4 liens à 100 Mbps ne donne un débit que de 100 Mbps) et non un cumul comme pour LACP ou FEC Une fois les *trunks* fonctionnels, nous nous sommes penchés sur les protocoles proposés par les commutateurs. Ces commutateurs permettent trois implémentations statiques et une seule implémentation dynamique en fonction du protocole choisi entre FEC[2] et LACP présenté ci après.

5.3.3. Protocoles d'agrégation

Les protocoles en question sont : LACP défini par la norme IEEE 802.3ad et FEC. Nous avons choisi d'implémenter LACP pour deux raisons : la première, c'est le protocole suggéré par le cahier des charges. La seconde est que LACP est une norme, non propriétaire et donc à priori présente sur tous les commutateurs, toutes marques confondues. En prévision d'une évolution possible du matériel actif, nous avons donc opté pour LACP.

5.3.4. *Port trunking* dynamique ou statique ?

Après avoir choisi LACP comme protocole d'agrégation, nous avons dû choisir entre une implémentation dynamique ou statique du protocole. Le *Dynamic LACP* permet une auto configuration des *trunks* lorsque les ports sont configurés comme *active* ou *passive lacp* (il faut qu'au moins un des deux périphériques ait les ports en *active lacp*). Dans cette configuration, nous ne pourrions plus utiliser un port particulier pour être moniteur (*monitoring port*[3]). Nous ne pourrions pas implémenter un algorithme autre que le STP (pas de RSTP possible par exemple). De même, si nous voulons mettre un lien d'agrégation dynamique en relation avec un VLAN[4] autre que le VLAN par défaut, nous ne pourrions pas faire de VLAN statique : il faudrait mettre en place GVRP[5] (implémentation dynamique des VLANs). De plus n'ayant que trois agrégations à faire, nous nous sommes permis de mettre en place un *trunk lacp statique*. Nous avons donc défait les *trunks* mis en place puis refaits ainsi en activant le mode LACP:

```
HPswitch(config)# trunk <liste des ports> <trk1 | ... | trk6>  lacp
```

5.4. Bilan

Une fois nos liens correctement mis en place, nous avons voulu tester notre solution. Le test de la redondance des liens a été trivial. Il nous a suffi de débrancher un câble pour nous assurer que les flux étaient bien répartis sur les trois autres liens. Pour tester l'augmentation du débit, test moins évident, nous avons pris en compte une machine du côté de hp4 dotée d'une carte gigabit et quatre machines du côté de hp1 doté de cartes 100 mbps. Nous avons ainsi formé quatre couples SA/DA[6] puis avons chargé au même moment les liens avec iperf

[2] *Fast EtherChannel*, protocole propriétaire cisco.
[3] Port où transitent toutes les informations pour une surveillance du trafic.
[4] *Virtual Lan Area Network* ou réseau locaux virtuels.
[5] *GARP Vlan Registration Protocol*, avec *GARP* pour *Generic Attribute Registration Protocol*.
[6] Sender Address / Destination Address

(une application client serveur permettant de mesurer les débits). Nous avons ainsi pu constater que la machine du côté de hp4 pouvait communiquer à environ 300 mbits, plus que les 100 mbps autorisés par une liaison inter commutateurs, maximum théorique d'une liaison unique ! Nous avons, pour mémoire, cherché à avoir plus d'informations sur le protocole FEC afin d'établir des comparaisons avec LACP, mais le protocole étant propriétaire, nous n'avons eu accès qu'au livre blanc de Cisco et aux spécifications du protocole. Les spécifications étant similaires à celles de LACP, nous n'avons pas constaté en pratique, de différence.

6. Les réseaux locaux virtuels

6.1. Qu'est ce qu'un réseau local virtuel

6.1.1. Une abstraction de la topologie physique

Les réseaux locaux virtuels, VLAN pour *virtual lan area network*, permettent de regrouper un ensemble de machines de façon logique, indépendamment de l'emplacement physique qu'elles ont dans le réseau. Le regroupement de ces machines est basé sur différents critères tels que l'adresse mac, le numéro de port, le protocole spécifique, etc. Ce sont autant de critères qui permettent d'identifier des types différents de VLAN. Ainsi, nous pouvons distinguer trois types de VLANs :

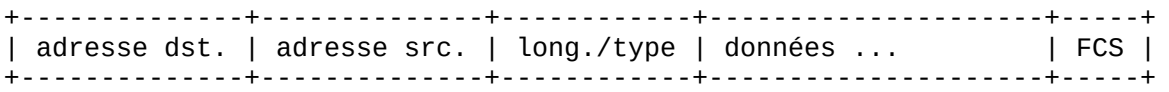
- Le premier niveau est le VLAN par port, le réseau local virtuel est défini en fonction des ports du commutateurs, un inconvénient majeur est qu'une station se déplaçant implique une modification de la configuration du port auquel elle était associée et du port auquel elle s'associe.
- Le deuxième niveau est le VLAN par adresse ethernet ou cette fois ci ce sont les adresses ethernet des machines qui permettent de déterminer leur appartenance au VLAN. Il est décrit comme plus souple que le niveau précédant. Il est en effet inutile de savoir où se trouve la machine dans le réseau, sur quel port ou sur quel commutateur puisqu'elle est identifiée par son adresse ethernet unique.
- Le troisième niveau est soit un VLAN par sous réseaux, soit un VLAN par protocole. Dans le premier cas nous gagnons en souplesse puisque les adresses sont identifiées par une adresse IP mais nous perdons en performance car les analyses des paquets doivent être plus précises. Le second type de VLAN permet de regrouper toutes les machines communicant avec un même protocole.

6.1.2. De l'utilité des VLANs

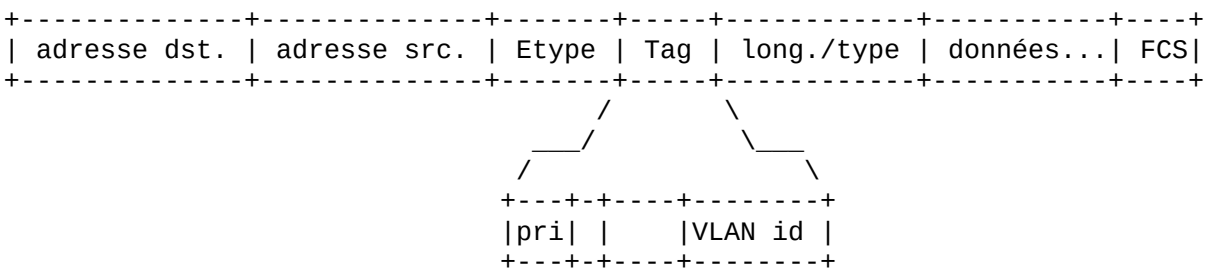
Nous pouvons citer premièrement une augmentation de la sécurité. Passer d'un réseau virtuel à un autre demande des fonctionnalités de routage ; sans ces dernières il est difficile d'accéder à un VLAN auquel on n'appartient pas, les VLANs sont alors étanches. Nous pouvons également citer une meilleure gestion de la bande passante. Dans le cas particulier d'une diffusion d'une machine vers tout le réseau (broadcast), toutes les machines sont atteintes, même celles qui ne sont pas du tout concernées. Cloisonner ces machines avec des VLANs permet donc d'éviter le gaspillage de bande passante en limitant la diffusion aux machines d'un même réseau virtuel. Confiner le trafic au sein d'un VLAN permet d'augmenter les performances du réseau. Cependant, l'évolution du parc implique une configuration supplémentaire selon le type de VLAN mis en place. Enfin nous pouvons indiquer que les VLANs permettent une limitation du domaine de broadcast ainsi qu'une segmentation des flux.

6.1.3. Communication inter VLANs

Nous avons énoncé le routage comme moyen pour « franchir l'étanchéité » des VLANs. Considérés comme des réseaux locaux (virtuels) le routage se fait de la même façon. L'échange des données se fait via un élement actif, routeur spécialisé ou une simple machine. Plusieurs scénarios existent. Le routeur doit évidemment avoir une patte dans chaque réseaux expéditeur et destinataire. Lorsque nous utilisons une machine pour router ses paquets, il faut par exemple utiliser deux interfaces (deux cartes réseaux) pour appartenir aux deux VLANs et ainsi pouvoir router aisément les paquets de façon classique d'un réseau vers l'autre. La trame Ethernet routée est alors la suivante :



Nous pouvons également utiliser une propriété des VLANs. Il est permis d'effectuer ce qui est appelé un *tag* : une machine peut appartenir à plusieurs VLANs à la fois. Il faut alors qu'elle possède le module 802.1q pour que ce « taggage » puisse se faire. Une seule interface physique est suffisante, les autres interfaces accédant aux autres VLANs restant logiques. Les trames échangées sont alors d'un format légèrement différent :



Le champ *Etype* est sur douze bits, également appelé *Tag protocol identifier (TPID)*, il permet d'identifier le protocole de la balise insérée. Dans le cas du 802.1q, le champ a la valeur 0x8100.
Le champ *priority* fait référence au standard IEEE 802.1p. Il est sur trois bits et permet de fixer un niveau de zéro à sept pour différencier l'importance des trames entre VLANs (qualité de service). Le champ CFI pour *Canonical format identifier* sur un bit permet d'assurer la compatibilité entre les adresses mac Ethernet (valeur égale à un) et tocken ring. Enfin le VLAN id, sur douze bits permet théoriquement de coder 4096 VLANs. En pratique, 4094 VLANs supplémentaires peuvent être pris en charge par l'adressage de la trame, parce que le 0 n'est pas utilisé et le VLAN 1 est celui par défaut, non modifiable. Nous pouvons souligner que le champ FCS est recalculé à chaque nouvelle insertion d'une étiquette dans la trame Ethernet.

6.1.4. L'implémentation du 802.1q

Le standard 802.1q est encore récent et non implémenté dans tous les systèmes d'exploitation ou par tous les équipementiers. Dans le cadre du stage les machines devant être « tagguées » étaient sous linux (fedora core). Toutes les machines linux dotées d'un noyau inférieur au 2.4.14 doivent être « patchées » puis recompilées pour être en mesure d'interpréter les trames « tagguées ». Il s'agit ensuite, à l'aide de l'utilitaire *vconfig* de configurer les interfaces

souhaitées. Par exemple, pour attribuer une interface logique au VLAN 5 sur l'interface physique eth0, il faut entrer après avoir chargé le module (*modprobe 8021q*) :

```
# vconfig add eth0 5
```

Il ne reste plus qu'à attribuer à cette interface l'adresse ip donnée au réseau virtuel :

```
# ifconfig eth0.5 192.168.0.2 netmask 255.255.255.0
```

Les versions Windows 2000 server supportent le standard 802.1p (priorité des trames) mais n'implémentent pas nativement le 802.1q. N'ayant pas de machines Windows à « tagguer », nous n'avons pas examiné la question plus en profondeur. Il s'agirait cependant d'utiliser les utilitaires fournis avec les pilotes Ethernet ou de créer son propre pilote « tagguant » les trames si le pilote n'existe pas pour la carte.
Encore une fois, il faut retenir que les équipements terminaux doivent pouvoir traiter le standard 802.1q (*802.1q compliant*) pour pouvoir interpréter des trames tagguées. Un équipement terminal n'intégrant pas le traitement du 802.1q dans sa pile IP ne doit pas être raccordé à un port taggué.

6.2. Scénario pour le CERMA

6.2.1. Les différents VLANs

Le CERMA souhaite mettre en place trois VLANs. Le matériel à disposition permet de faire uniquement des VLANs par port. C'est donc le type de VLANs que nous avons créés. Un VLAN destiné aux impressions dans lequel vont se trouver les imprimantes et les serveurs d'impressions, ce qui permet de séparer les flux d'impressions du reste du trafic. Un autre pour l'administration dans lequel se trouvent les éléments actifs et autres machines de monitoring. Le dernier regroupant le reste des machines du laboratoire. Aucune communication ne sera possible des machines utilisateurs vers le VLAN administration ou impressions. Les machines utilisateurs impriment via deux serveurs d'impressions (sous linux) qui appartiennent à la fois au VLAN d'impression et au VLAN des utilisateurs, et implémentent donc le module 802.1q.

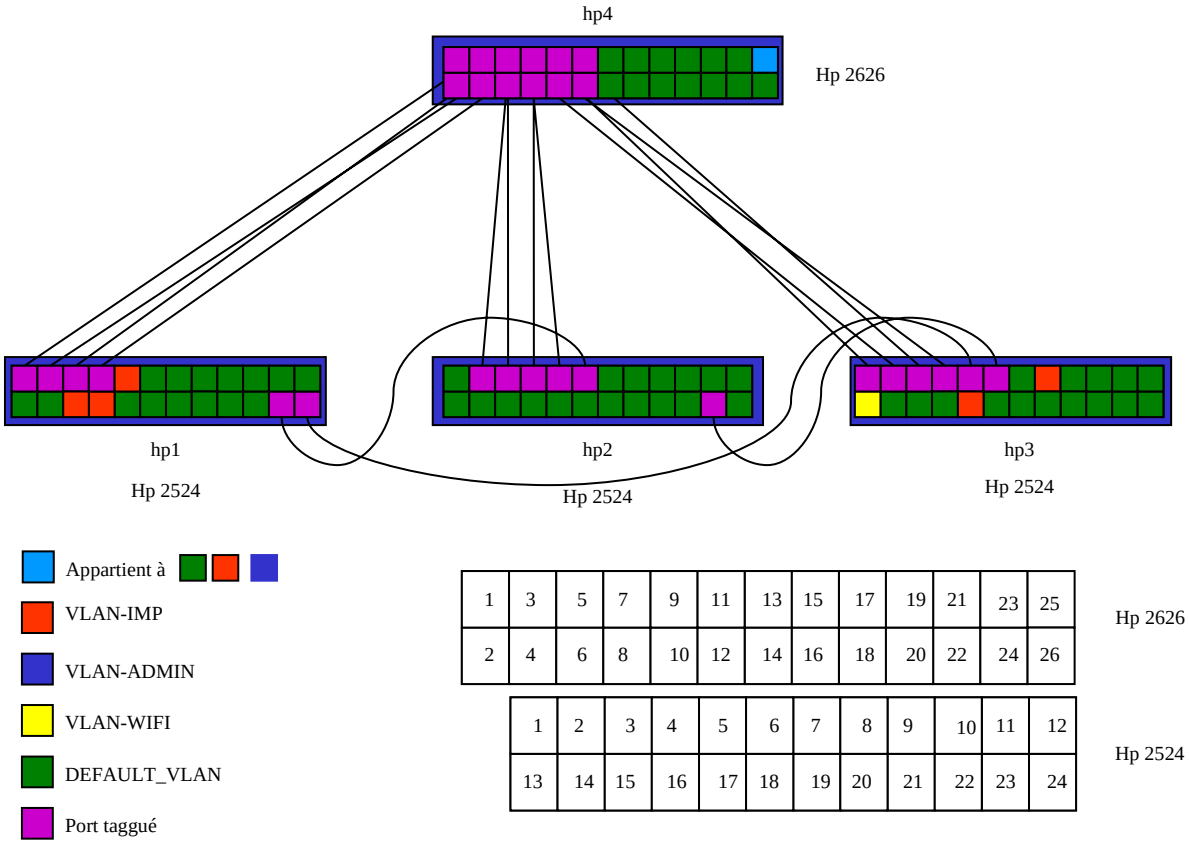
6.2.2. Le problème du DHCP

Un problème rencontré durant l'implémentation de la solution VLAN aura été la gestion de l'adressage dynamique des machines situées sur les VLANs (via le DHCP[7]). En effet comment faire pour que le serveur DHCP puisse alimenter tous les VLANs ? Comme énoncé plus haut, plusieurs solutions sont possibles : tagguer le serveur DHCP de façon à ce qu'il soit dans tous les VLANs ou faire du *DHCP-relay*. Cette solution consiste à router les requêtes DHCP des clients vers le serveur et les réponses du serveur vers les clients et ce, même si le serveur ne se trouve pas dans le domaine de diffusion (dans le même VLAN). Le routage et le *DHCP-relay* sont deux fonctionnalités que possède le *hp procurve* 2626. C'est une solution plus élégante que d'obliger le serveur DHCP à figurer dans tous les VLANs. Cependant, dans le cadre du CERMA, la machine serveur DHCP est également le serveur d'impression, déjà tagguée dans deux VLANs et également présente dans le VLAN par défaut contenant toutes les machines utilisateurs. De ce fait, l'ensemble des machines est donc alimenté par le serveur DHCP. La solution *DHCP-relay* à quand même été testée avec succès (cf. le mode opératoire

[7] *Dynamic Host Control Protocol*, protocole permettant d'attribuer les adresses ip dynamiquement.

pour la mise en place du *DHCP-relay* en annexe) lorsqu'il nous a été demandé de rajouter un quatrième VLAN, le VLAN wifi, contenant uniquement la borne d'accès. La borne d'accès pouvant elle-même faire office de serveur DHCP pour les clients, nous n'avons pas conservé la configuration, bien que fonctionnelle. En revanche, la fonction de routage (*ip routing*) des flux du VLAN wifi vers le VLAN utilisateurs pour que les utilisateurs du sans fil ait accès au réseau local et à Internet notamment.

Le schéma ci-dessous présente la configuration des ports sur les différents commutateurs.



Implémentation sur le matériel du CERMA

La manipulation en détails et les différentes étapes (*step by step*) pour mettre en place les VLANs sont présentées en annexe (mode opératoire de l'implémentation des VLANs). Nous donnons ici les principales étapes avec les lignes de commandes utiles pour la compréhension du concept.

6.2.3. Allocation du nombre de VLANs dans le réseau

La première étape de la mise en place des VLANs est de renseigner sur chacun des commutateurs le nombre de VLANs du réseau. Nous avons donc indiqué 4 comme nombre des VLANs du réseau du CERMA. Nous rappelons pour mémoire qu'il s'agit d'un VLAN pour l'administration, d'un VLAN pour l'impression, d'un VLAN pour les utilisateurs et d'un dernier VLAN pour le wifi. La commande qui suit a donc été lancée sur chaque commutateur :

```
Hp(config) # max-vlan 4
```

6.2.4. Création des VLANs sur les commutateurs

Une fois le nombre de VLANs renseigné, nous avons créé, toujours sur chaque commutateur, les différents VLANs à l'aide des commandes suivantes :

```
Hp(config) # VLAN 11 name VLAN-IMP
Hp(config) # VLAN 12 name VLAN-ADMIN
Hp(config) # VLAN 13 name VLAN-WIFI
```

Nous soulignons qu'il est indispensable que les identifiants numériques des VLANs soient les mêmes sur chaque commutateur. Nous rappelons également que si le VLAN utilisateur (*default VLAN*) n'est pas renseigné, c'est que l'identifiant n'est pas modifiable (égale à 1). Nous l'avons cependant renommé en « VLAN-autres ».

6.2.5. Paramétrage des VLANs

Le paramétrage des VLANs consiste en l'affectation des ports dans les différents VLANs. Les ports doivent appartenir à un VLAN particulier (*default VLAN*... par défaut). Pour des questions de commodité d'administration, tous les ports terminaux sont dans le *default VLAN*. Nous devons également renseigner leur statut. C'est-à-dire, indiquer si ils sont taggués (statut *taggued*) ou non (statut *untaggued*) lorsqu'il appartient à un VLAN. Un port ne peut pas avoir le statut *untaggued* dans plusieurs VLANs. Lorsque le port n'appartient pas à un VLAN, on lui renseigne le statut *no* (le statut *forbid* est pour l'implémentation dynamique des VLANs). Les lignes de commandes permettant de paramétrer ces options sont :

```
Hp(config)# VLAN id tagged <liste-port>
Hp(config)# VLAN id untagged <liste-port>
```

Ainsi, par exemple, toutes les imprimantes sont placées en *untagged* dans le VLAN d'identifiant 11 et la machine tage est paramétrée en *taggued* dans les VLANs d'id 11 et 12, *untaggued* dans le VLAN 1. Il est moins coûteux d'échanger des trames non étiquetées dans le VLAN où la machine devra le plus souvent communiquer, d'où l'activation du *untagged* dans le VLAN utilisateurs. Un paramètre important est de permettre aux liens d'agrégation de pouvoir transporter des trames tagguées. C'est la raison pour laquelle tous les trunks sont paramétrés comme *taggued* pour permettre au trafic étiqueté de transiter d'un commutateur à un autre.

6.2.6. Choix d'un VLAN primaire

Le VLAN primaire est le VLAN qui s'occupe de gérer les informations communes à tous les VLANs. Certaines fonctionnalités comme le DHCP pour ne citer que celle qui nous intéresse ne se réalisent que par VLAN. Par exemple, le commutateur lit les réponses DHCP dans le VLAN primaire. Par défaut, c'est le *default VLAN* qui est primaire. Nous avons cependant préféré indiquer le VLAN administration comme étant le VLAN primaire bien que conceptuellement il n'y ait pas de différence. Ce choix est bien sur indépendant du commutateur, mais nous avons effectué le changement sur les quatre commutateurs afin de rester cohérent. Nous avons donc utilisé la commande :

```
Hp(config) # VLAN 12 primary-VLAN
```

6.2.7. Identification des réseaux virtuels

Comme tout réseau (virtuel), nous devons identifier nos différents VLANs. Nous avons donc définis arbitrairement des adresses pour les différents VLANs, toutes en classe C non routable : 192.168.10.0 pour le VLAN utilisateurs, 192.168.11.0 pour le VLAN d'impression, 192.168.12.0 pour le VLAN d'administration et 192.168.13.0 pour le VLAN wifi.
Les commandes qui nous ont permis cette identification sont, lorsque nous avons souhaité pour les premiers tests les renseigner en manuel sur tous les commutateurs (sans le DHCP et le *DHCP-relay* dans notre cas) :

```
Hp(config) # VLAN 10 ip address 192.168.10.0/24
Hp(config) # VLAN 11 ip address 192.168.11.0/24
Hp(config) # VLAN 12 ip address 192.168.12.0/24
Hp(config) # VLAN 13 ip address 192.168.13.0/24
```

Puis, plus simplement en optant pour une configuration dynamique à l'aide du DHCP (après activation du *DHCP-relay*, voir ci après) :

```
Hp(config) # VLAN 10 ip address DHCP-boot
Hp(config) # VLAN 11 ip address DHCP-boot
Hp(config) # VLAN 12 ip address DHCP-boot
Hp(config) # VLAN 13 ip address DHCP-boot
```

Note : ce sont les adresses MAC qui permettent d'identifier les VLANs pour leur affecter les adresses réseaux dans le fichier *DHCP.conf* du serveur DHCP.

6.2.8. Activation et configuration du routage

Comme indiqué dans la section « scénario pour le CERMA » ci-avant, nous avons cherché dans un premier temps à router le trafic du VLAN wifi vers le VLAN utilisateurs. Dans un second temps, le routage est une condition nécessaire au fonctionnement du *DHCP-relay*, cf. ci après. Pour effectuer donc le routage à l'aide du commutateur *hp procurve* 2626 (hp4), nous activons simplement la fonction *ip routing* :

```
Hp4(config) # ip routing
```

Lorsque les trames sont envoyées sur hp4, le commutateur est capable de les router vers les adresses réseaux renseignées avant.

6.2.9. Activation et configuration du *DHCP-relay*

Il s'agit ici de permettre aux machines n'étant pas dans le domaine de diffusion du serveur DHCP d'émettre des requêtes et d'obtenir une adresse ip dynamiquement. Le principe est d'indiquer que tel VLAN adressera ses requêtes DHCP à tel adresse, celle du serveur DHCP.
La commande pour activer le *DHCP-relay* est :

```
Hp4(config) # DHCP-relay
```

Nous renseignons ensuite l'adresse du serveur DHCP associé à un VLAN comme suit :

```
Hp4(config) # VLAN 12 ip helper-address 192.168.10.3
```

Où 192.168.10.3 est l'adresse du serveur DHCP et id l'identifiant du VLAN. Les machines du VLAN administration sont donc capables d'obtenir une adresse ip dynamiquement. Nous précisons que seul hp4 (procurve 2626) parmi l'ensemble des commutateurs du CERMA, implémente cette fonctionnalité.

6.3. Bilan

La mise en place des VLANs n'a pas été triviale. Si elle nous paraît maintenant beaucoup plus abordable, nous avons eu du mal à en saisir rapidement tous les concepts. Nous avons par exemple perdu du temps lors du « tagguage » d'une machine dans plusieurs VLANs car nous n'avions pas considéré que l'équipement terminal devait également être « *802.1q compliant* » afin de pouvoir interpréter et envoyer des trames comportant des étiquettes. Une fois ce concept bien intégré nous avons mieux compris ce qui ce passait pour faire en sorte qu'une machine soit dans plusieurs VLANs. C'est ainsi que nous avons dû rajouter une carte réseau à une machine linux, dont le noyau était inférieur au 2.4.14 et non recompilé avec le module 802.1q, afin qu'elle puisse se trouver effectivement dans deux réseaux (virtuels) différents. Le second souci a été également conceptuel. Il nous a fallu un certain temps pour comprendre qu'un réseau, même virtuel, devait être identifié par une adresse ip si nous voulions faire du routage (couche 3 du modèle OSI). Ceci, indépendamment des ports qu'il regroupe dans notre cas. Une fois ces concepts assimilés les VLANs deviennent effectivement simple de configuration et nous comprenons effectivement mieux pourquoi notre réseau virtuel peut être paramétré et utilisé comme un réseau réel.

7. Le *spanning tree*

7.1. Qu'est ce que le spanning tree ?

Le STP pour *spanning tree protocol* est définie par l'IEEE dans le document 802.1d. Sa principale utilité est de s'assurer qu'il n'y a pas de boucle dans un contexte de liaisons redondantes entre des matériels de couche 2. L'algorithme détecte les boucles en question et les désactive. Ces liens désactivés servent de liens de backup et ne sont débloqués que si les liens actifs tombent en panne. Il n'y a ainsi qu'un seul chemin actif à une date donnée entre deux équipements actifs.

7.2. Le fonctionnement du spanning-tree

Le STP crée un chemin sans boucle basé sur un chemin le plus court. Ce chemin est établi en fonction de la somme des coûts des différents liens le constituant (liens inter switchs). Nous comprenons donc qu'établir un chemin sans boucle peut entraîner que des ports soient bloquées et d'autres pas. Le protocole échange fréquemment des trames particulières appelées BPDU[8] afin d'adapter la topologie à d'éventuelles modifications : l'arbre minimal doit être maintenu.

7.2.1. Le commutateur racine

Le spanning-tree (ou arbre recouvrant) possède comme tout arbre une racine. C'est le point « central ». Une élection est faite pour déterminer le commutateur qui sera la racine. Le commutateur élu est celui qui possède le plus petit identifiant. Cet identifiant est composé de deux parties, une priorité (sur deux octets) et l'adresse mac (sur six octets). Pour culture, la priorité définie par le standard 802.1d est d'une valeur de 32768 par défaut, ce sont des multiples de 4096 sur 16 bits. Cette priorité peut être modifiée au niveau du commutateur (*switch priority*). Sur un commutateur racine (root), tous les ports sont activés, ils émettent et reçoivent.

7.2.2. Le port racine

Chaque commutateur doit sélectionner un port racine (root port) qui désigne le chemin le plus court (le moins coûteux) vers le commutateur racine. Ce port racine est dans un état que l'on nomme *forwarding*.

7.2.3. Les valeurs par défauts sur les commutateurs

Nous expliquons ici le rôle des valeurs par défaut accessibles sur le commutateur dans le protocole du *spanning tree*. L'age maximal (*max age*) de 20 secondes par défaut est le temps maximal pendant lequel on considère qu'une trame BPDU est valide. Le temps de *forwarding* (*forward delay*) de 15 secondes par défaut est le temps de passage d'un état "listening" à "learning" et d'un état "learning" à "forwarding" ; cette valeur est transportée dans les BPDUs, afin de rester homogène, il est conseillé de ne la changer qu'au niveau de la racine. La fréquence d'envoi des trames BPDUs appelées trames Hello (*Hello time*) est de 2 secondes.

[8] *Bridge protocol data unit*

7.3. Les différents états des ports configurés par le spanning-tree

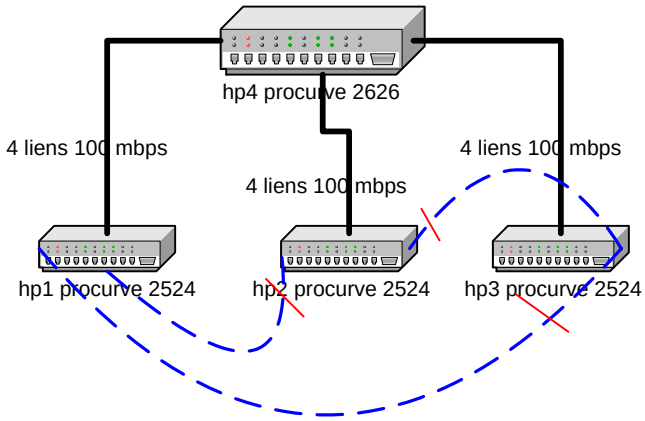
Les ports peuvent se retrouver dans cinq états (blocking, listening, learning, forwarding, disabled) dont nous énonçons les propriétés :

- Dans l'état blocking, le port bloqué n'émet et ne reçoit pas de trames, la table de forwarding ne doit pas être mise à jour. Seul les trames BPDU sont traitées pour vérifier la topologie ;
- Un port dans l'état listening n'émet pas et ne reçoit pas de trames, la table de forwarding n'est pas tenue à jour. Les trames BPDU sont reçues, traitées et émises ;
- Un port dans l'état learning se prépare à la phase de pontage et même s'il bloque toutes les trames, il met à jour sa table de forwarding ;
- Un port en mode forwarding transfère les trames de et vers le segment, met à jour sa table de forwarding et traite les BPDU ;
- Enfin, dans l'état disabled, le port est considéré comme physiquement non opérationnel (problème physique), il ne participe pas au *spanning tree*, ne traite aucun paquet, pas même les BPDU.

7.4. Scénario pour le CERMA

7.4.1. L'arbre recouvrant pour le CERMA

Des liens redondants ont été rajoutés entre hp1, hp2 et hp3.



Le *spanning tree* implémenté doit bloquer les liens redondants (en pointillés). Pour ce faire, la somme des coûts des liens des *trunks* reliant deux commutateurs à hp4 doit être inférieure au coût du lien reliant directement ces deux commutateurs. Précisons que ces liens entre deux commutateurs seront automatiquement débloqués si les liens d'agrégation (le *trunk* entier !) venait à tomber entre deux commutateurs. Enfin, les liens doivent être « taggués » sur tous les VLANs pour pouvoir faire circuler le trafic de tous les VLANs.
Les commutateurs à notre disposition permettent d'implémenter le *spanning tree* via trois protocoles : STP (802.1D), RSTP (802.1w) et MSTP (802.1s)[9].

[9] *Spanning Tree Protocol, Rapid reconfiguration Spanning Tree Protocol, Multiple Spanning Tree Protocol.*

7.4.2. Différence entre STP et RSTP

Le STP est l'implémentation du protocole permettant de bloquer les liens redondants, une fois activé. Le RSTP est une version (compatible avec STP) qui permet l'établissement de la cartographie du réseau par le protocole plus rapide. La version STP met plus de temps à trouver les différents chemins (liens) et à en sélectionner les plus efficaces. MSTP est une version qui autorise la mise en place de plusieurs instances du *spanning tree* dans le réseau. Nous imaginons alors une application directe : nous pourrions faire une instance de *spanning tree* par vlan. Mais nous avons constaté par la suite que seul un des quatre commutateurs permettait cette manipulation.

7.5. Implémentation sur le matériel du CERMA

La manipulation en détails et les différentes étapes (*step by step*) pour mettre en place le protocole RSTP sont présentées en annexe (mode opératoire de l'implémentation du RSTP statique). Nous donnons ici les principales étapes avec les lignes de commandes utiles pour la compréhension du concept.

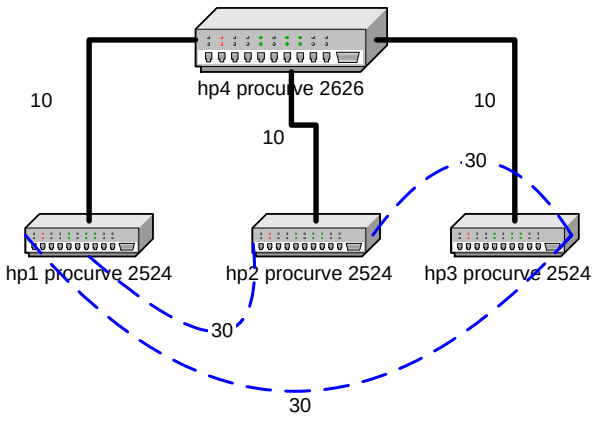
7.5.1. Activation du RSTP

Pour les raisons de performances entre les deux algorithmes RSTP et STP énoncés plus haut et parce que les commutateurs que le CERMA possède le permettent, nous avons opté pour l'implémentation du RSTP. Nous paramétrons le protocole sur tous les commutateurs via la ligne de commande :

```
Hp4(config) # spanning-tree protocol-version rstp
```

7.5.2. Configuration du protocole

Nous laissons les paramètres du protocole (cf. propriétés dans le paragraphe sur le fonctionnement du *spanning tree*) par défaut et nous nous penchons plus précisément sur les paramètres *port cost*, *edge* et *priority*. Le premier nous permet de créer l'arbre minimal recouvrant, le deuxième indique que la connexion à l'autre bout est bien un commutateur (la valeur doit être dans ce cas égale à *No*). Enfin le troisième permet de favoriser des liens inter commutateurs à coût égal pour établir une qualité de service. Nous souhaitons que le trafic passe par les liens d'agrégation, il nous faut donc que la somme des coûts sur les liens entre deux commutateurs (parmi hp1, hp2 et hp3) et hp4 soit inférieur à la liaison directe entre les deux commutateurs. Nous pouvons donc placer les coûts comme suit :



Ainsi, puisque 10 + 10 = 20 < 30, le trafic passera par les liens pleins (agrégation) et les liens bloqués seront les liens en pointillés.

Pour effectuer cette manipulation dans la configuration des commutateurs, il faut entrer la ligne de commande suivante :

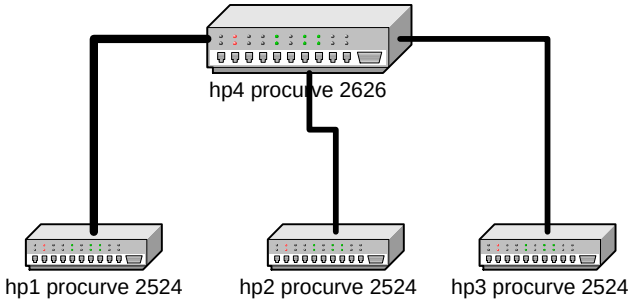
```
Hp4(config) # spanning-tree <port-list> path-cost <1-2000000>
```

Des valeurs sont indiquées dans la documentation concernant le coût à attribuer en fonction de la vitesse du port. Le tableau ci-dessous récapitule les plages de valeurs proposées :

Type du port	Coût du lien avec STP	Coût du lien avec RSTP
10 Mbps	100	2 000 000
100 Mbps	10	200 000
1000 Mbps	5	20 000

7.5.3. La racine de l'arbre

D'après les configurations précédentes l'arbre obtenu prendrait cette forme :



Intuitivement la racine de l'arbre serait le commutateur hp4. Nous avons donc opté pour le définir en racine. Ceci à l'aide du paramètre *switch priority* que nous mettons au minimum afin d'obtenir la priorité la plus importante.

```
Hp4(config) # spanning-tree priority <0-15>
```

Par précaution nous définissons aussi une seconde racine, de secours, au cas où hp4 tomberait en panne. Les liens redondants seraient alors activés et une nouvelle racine devra être élue. Nous suggérons donc hp2 comme nouvelle racine, de façon arbitraire, en lui indiquant une priorité moins importante que celle de hp4 et plus importante que celles des commutateurs hp1 et hp3.
Note : 0 est la priorité la plus forte.

7.6. Bilan

Bien qu'ayant étudié le *spanning tree* en détail en licence informatique, il nous a fallu un certain temps pour nous rendre compte que les problèmes que nous avions lors d'une première tentative de l'activation du *spanning tree* étaient dus au fait que nous avions mal construit l'arbre. Sur le papier, nous avions l'arbre correct mais nous n'avions pas tenu compte, lors de la configuration sur les commutateurs, de la somme des coûts des liens entre les commutateurs passant par la racine. Elle n'était donc pas inférieure au coût de la liaison directe entre les deux commutateurs, l'arborescence ne correspondait plus à ce que nous voulions obtenir.

8. Qualité de service

8.1. Qu'entend-on par qualité de service ?

La qualité de service (QoS) peut être définie comme la conformité d´un service à répondre aux exigences d´un client, qu´il soit externe ou interne. La difficulté est de mesurer précisément cette qualité de service. Il faut donc distinguer entre le service attendu (les besoins des utilisateurs), le service rendu et le service perçu. Nous comprenons alors que cette qualité de service englobe des paramètres très divers qui reposent sur un nombre important d'indicateurs. Respecter la qualité de service est par exemple aussi bien tenir compte des précautions à prendre sur le temps d'intervention sur un serveur que les variations de latence des paquets (notion de gigue).

8.2. Scénario pour le CERMA

Il s'agit de définir quels paramètres nous prenons en compte pour implémenter une qualité de service. Dans le cadre du réseau du CERMA, nous optons pour donner des priorités différentes selon que le trafic appartienne au VLAN impression ou au VLAN utilisateurs par exemple. Ainsi, le trafic du VLAN d'impression sera « plus important » que le trafic du VLAN d'administration qui sera lui-même plus prioritaire que le trafic du VLAN comportant l'ensemble des utilisateurs. Une première contrainte est le matériel : les trois commutateurs *hp procurve* 2524 ne sont pas capables d'implémenter une qualité de service. En revanche, le commutateur 2626 est capable de gérer une qualité de service dans un environnement de VLANs « taggués » ou non. Ce dernier commutateur dispose également de deux ports gigabit que nous allons privilégier. Une question peut alors être posée : que se passe-t-il lorsque le trafic sur un port gigabit d'un VLAN A moins prioritaire qu'un VLAN B est en concurrence avec le trafic par un port de 100 Mbps appartenant à ce VLAN B ? Il existe une notion de précédence qui permet de graduer les différents types de QoS. L'ordre de priorité sur les *HP Procurve* est tel que la QoS sur les ports est plus importante que celle sur le trafic des VLANs.

8.3. Implémentation sur le matériel du CERMA

8.3.1. Priorité des VLANs

Pour ce faire, au moins un VLAN doit être taggué sur le réseau afin d'identifier le trafic prioritaire. Nous pouvons noter également que ce type de QoS (basé sur les identifiants des VLANs) ne peut pas être configurer lorsque les VLANs sont dynamiques (GVRP). La syntaxe de la commande est simple et consiste à indiquer la priorité aux différents VLANs.

```
Hp4(config) # VLAN <identifiant> qos priority <0 -7>
```

Plus le chiffre est proche de 0, plus la priorité est élevée. Une fois les priorités placées, il est possible de consulter les différentes priorités via la commande `show qos vlan-priority`.

8.3.2. Priorité des ports gigabits

De façon similaire, l'affectation de la QoS sur les ports se fait en leur affectant une priorité via la commande suivante :

```
Hp4(config) # interface <port list> qos priority <0 -7>
```

Plus le chiffre est proche de 0, plus la priorité est élevée. Une fois les priorités placées, il est possible de consulter les différentes priorités via la commande `show qos port-priority`. Nous soulignons encore une fois que cette QoS à un niveau de priorité supérieur à la QoS affecté sur le trafic des VLANs.

8.4. Bilan

Nous avons vu ici un aspect minimaliste de ce que peut apporter la qualité de service. En effet, la qualité se service permet de faire beaucoup plus. Nous aurions pu par exemple dans la gamme du dessus, répartir le trafic d'un VLAN donné, d'un protocole particulier sur des liens spécifiques. Cependant la solution proposée permet de donner plus d'importance à certaines machines (Tage notamment, à la fois serveur DHCP et serveur d'impression) puis à certains VLANs. De plus la segmentation des flux d'administrations permet d'éviter que les messages de configuration du protocole du *spanning tree* soient en concurrence avec le trafic d'impression ou celui des utilisateurs.

9. Bilan du stage

Le stage que nous avons réalisé au CERMA a été très complet, tant du point de vue technique, où il nous a fallu maîtriser le fonctionnement des commutateurs HP, que du point de vue théorique, où nous avons dû revoir nos connaissances sur le *spanning tree* par exemple mais aussi découvrir de nouveaux protocoles (LACP, FEC, CDP, ...) et appréhender de nouveaux concepts (VLAN) ou manipuler de façon plus ou moins avancée des notions importantes (QoS, routage). Il nous a fallu travailler sur un réseau en exploitation ce qui impliquait que chaque manipulation devaient avoir été préparées au préalable afin de causer le moins de désagrément possible auprès des utilisateurs. Nous avons dû trouver des solutions en adéquations avec les moyens dont nous disposions et tenir compte des besoins exprimés par l'administrateur réseaux (routage entre le VLAN wi-fi et VLAN autres, activation du *DHCP-relay*, mise en place d'une qualité de service). Durant le stage, nous nous sommes retrouvés confronter à des problèmes liés à la spécificité du réseau (problème de compatibilité Mac OS9/ *spanning tree*) qui nous ont obligé à chaque fois à nous adapter. Les solutions que nous avons trouvées ne sont peut être pas les plus performantes mais elles sont celles qui conviennent le mieux au réseau actuel du CERMA. L'apport de ce stage nous sera très bénéfique dans le futur de par sa richesse d'enseignements et d'autre part parce qu'il nous aura permis de mieux cerner les aspects professionnels du poste d'administrateur réseau au sein d'une entreprise, notamment de prendre conscience de la relation qui le lie avec les usagers du réseau. En effet celui-ci est le garant de la qualité et de l'intégrité du réseau placé sous sa responsabilité ; responsabilité parfois un peu ingrate : lorsque tout fonctionne l'administrateur reste inconnu, lorsqu'un problème survient il devient la vedette malheureuse de l'entreprise☺.

10. Quelques références

Documentation des commutateurs HP :
http://www.hp.com/rnd/support/manuals/2650_6108.htm
http://www.hp.com/rnd/support/manuals/23xx_25xx.htm
http://www.hp.com/rnd/support/faqs/ProCurve-Manager.htm

Configuration des liens LACP par Cisco :
http://www.cisco.com/en/US/tech/tk389/tk213/technologies_configuration_example09186a0080094470

Fonctionnement des *trunks* :
http://www.ieee802.org/3/trunk_study/tutorial/index.html

Une bonne explication de ce que sont les réseaux locaux virtuels et un chapitre intéressant sur les bases de la construction de VLAN :
Les réseaux locaux virtuels, de Gilbert Held, *InterEditions*

Définition et fonctionnement des VLANs :
http://www.awt.be/web/fic/index.aspx?page=fic,fr,t00,015,002
http://net21.ucdavis.edu/newvlan.htm
http://www.urec.cnrs.fr/cours/Liaison/vlan/

Des informations concernant le routage inter VLAN :
http://www.linux-france.org/prj/inetdoc/cours/routage.inter-vlan/intro.vlan.html

Un TP qui détails pas à pas un exemple de configuration des VLANs (avec des *HP procurve* !) pour une bonne compréhension du concept :
http://etudiant.univ-mlv.fr/~jlegra02/fichiers/Reseau/TP6_VLAN.pdf

Des informations sur le *spanning tree* :
http://www.reseaucerta.org
http://www.cis-consultants.com/txt/technotes/STP.html

Quelques transparents qui précisent la notion de routage IP :
http://www.urec.cnrs.fr/cours/Reseau/routip1/sld001.htm