Réseau du Campus Jussieu

Domain Name System - Fonctionnement et Installation

Messagerie - Architecture et Installation

Pierre David
Laboratoire MASI
Université de Versailles – St Quentin en Yvelines
(Anciennement aux laboratoires LPTL-LPTHE)
email : Pierre.David@masi.ibp.fr

Jacky Thibault
Centre de Calcul Recherche
Université Pierre et Marie Curie
email : Jacky.Thibault@ccr.jussieu.fr

Version 1
1er avril 1993

Table des Matières

Chapitre 1

Introduction

Ce document est le guide de configuration du serveur de noms et de la messagerie. Ces services sont les services minimum pour utiliser le réseau du Campus Jussieu à partir de votre laboratoire.

Le chapitre suivant introduit un outil (*ftp anonyme*) mis à votre disposition afin de retrouver des informations utiles pour la suite.

Le troisième chapitre explique les principes de fonctionnement du *Domain Name System*, puis sa configuration dans les laboratoires.

Le dernier chapitre présente la messagerie et l'intégration du réseau du Campus dans le contexte international. Les diverses stratégies de configurations dans les laboratoires sont ensuite détaillées.

Enfin, les annexes donnent quelques éléments utiles.

Les auteurs seraient heureux de recevoir les commentaires et critiques, afin d'améliorer tant la forme que le fond.

Chapitre 2

Ftp anonyme

Au titre de l'assistance aux administrateurs des systèmes connectés au réseau du Campus Jussieu, un mécanisme de récupération des informations a été mis en place. Il s'agit du *ftp anonyme*.

2.1 Présentation du ftp anonyme

Le *ftp anonyme* est une vieille tradition sur le réseau *Internet*. Il s'agit d'autoriser un accès libre à un certain nombre de fichiers. Des milliards d'octets de programmes, de fichiers sources et d'autres informations sont disponibles sur des sites *ftp anonyme* du réseau *Internet*.

Pour utiliser un site fournissant le *ftp anonyme*, rien n'est plus simple. Faites `ftp` *site*. Lorsqu'on vous demande votre nom, rentrez `anonymous` (le nom de login `ftp` est aussi admis sur certains systèmes). Lorsqu'on demande votre mot de passe, donnez votre adresse électronique (par exemple `pda@jussieu.fr`). Il s'agit là d'une convention, une sorte de politesse à l'égard de ceux qui ont installé et maintiennent le site.

Une fois connecté, vous pouvez utiliser toutes les commandes de `ftp` habituelles : `cd`, `ls`, `get`, etc... Reportez-vous à la documentation de `ftp` fournie avec votre système.

2.2 Le ftp anonyme à Jussieu

Le site `ftp.jussieu.fr` (adresse IP 134.157.0.129) a été configuré en *ftp anonyme*. Sont disponibles, entre autres :

- l'intégralité des *Request For Comments* (documents définissant les standards de l'*Internet*) ;
- des versions de X11 compilées pour diverses architectures ;
- des patches pour des machines diverses ;
- etc.

Usez et abusez du *ftp anonyme* sur `ftp.jussieu.fr.`. Aidez à maintenir ce site en avertissant `pda@jussieu.fr` de ce que vous aimeriez y voir figurer.

Chapitre 3

DNS - Fonctionnement et Installation

3.1 Introduction

Le DNS (*Domain Name System*[1]) est un système distribué et cohérent de recherche d'informations sur le réseau *Internet* dont le réseau du Campus Jussieu fait partie. Ce système permet, entre autres, de retrouver le nom ou l'adresse de n'importe quel ordinateur sur le réseau *Internet*.

Ce chapitre se propose de présenter le fonctionnement et l'administration du DNS à Jussieu. Les principes d'utilisation sont traités dans [1] et sont supposés connus ici.

3.2 Historique du DNS

Cette section a pour but de présenter les principes de conception du DNS.

3.2.1 Conversion de nom en adresse IP et vice-versa

Les machines sur le réseau *Internet* communiquent avec des adresses IP[2]. Si cela va bien pour des machines, il n'en va pas de même pour des humains. Il nous faut des noms et pas des numéros.

Sur les systèmes Unix traditionnels, les conversions de noms d'ordinateurs en adresses IP sont effectuées à l'aide du fichier `/etc/hosts`.

Lorsqu'un utilisateur sur une machine `a` tape la commande `rlogin b`, la machine `a` consulte son fichier `/etc/hosts` pour connaître l'adresse IP de `b`. Lorsque la machine `b` reconnaît une demande de connexion, elle ne dispose que de l'adresse IP de `a` et, par conséquent, consulte son fichier `/etc/hosts` pour obtenir

[1] Certains mots ou expressions américains, en l'absence de traduction reconnue et acceptée, sont utilisés tels quels.
[2] Quatre nombres séparés par des points, par exemple 134.157.1.1

son nom. Elle peut alors autoriser ou non la connexion automatique en fonction des noms contenus dans les fichiers `/etc/hosts.equiv` et `.rhosts`.

3.2.2 Avant le DNS

Historiquement, dans le feu réseau ARPANET, où chaque machine pouvait accéder à toute autre, les ordinateurs disposaient d'un fichier `/etc/hosts` listant tous les autres ordinateurs du réseau.

Si ce schéma simple est efficace pour de petits réseaux, il n'en va pas de même pour un réseau à l'échelon mondial comme *Internet*, répertoriant aujourd'hui plus de 1 500 000 ordinateurs :

- au rythme d'ajout de nouvelles machines, la maintenance d'un tel fichier devient impossible ;
- la propagation de ce fichier vers tous les ordinateurs du réseau représente un trafic considérable ;
- la transition de gros ordinateurs vers des kyrielles de stations de travail rend le nombre d'interlocuteurs démesuré ;
- l'unicité des noms devient un problème aigu.

3.2.3 Principes de conception du DNS

Le DNS (voir [7][8][9][10]) a été conçu en 1987 pour palier ces problèmes. Le DNS est un mécanisme général et pas seulement un mécanisme de traduction de noms en adresses IP et vice-versa. C'est un système de nomage unique pour toutes les ressources (adresses de machines, services offerts, adresses de groupes, noms de réseaux, etc.). Le DNS, de par sa généralité, est facilement extensible. Par exemple, la définition originale a été complétée en 1989 (voir [11]) pour gérer les noms et caractéristiques des réseaux, et en 1990 pour le système Kerberos développé au MIT.

Le DNS repose sur un système de nomage hiérarchique comme décrit dans [1].

De plus, le DNS est un exemple d'application répartie sur l'ensemble du réseau *Internet*. Ceci permet, entre autres bénéfices, d'ajouter une machine sur le réseau du Campus Jussieu sans affecter le fonctionnement des machines au fin fond des Etats-Unis. Cependant, une machine du fin fond des Etats-Unis pourra profiter de cet ajout, et un utilisateur (s'il est autorisé bien entendu) pourra utiliser **`rlogin`** pour se connecter avec le vrai nom de la nouvelle machine. La modification est faite localement sur le Campus, et cette nouvelle machine devient partie intégrante du réseau *Internet*.

Cependant, cet aspect "distribué" du DNS pose le problème du trafic induit nécessaire pour obtenir l'information. C'est pour résoudre ce problème et ne pas saturer le réseau qu'un mécanisme de "cache" a été mis au point, afin de mémoriser les dernières informations obtenues. On évite ainsi des requêtes superflues lors la recherche ultérieure de ces mêmes informations.

Ce mécanisme de cache pose, à son tour, un problème : les informations qui y sont placées sont-elles cohérentes, sont-elles à jour ? Les concepteurs du DNS ont supposé que les informations ne changent pas trop rapidement, et ont fourni aux utilisateurs un moyen pour contrôler la vitesse de mise à jour. Ceci étant, il faut gérer un compromis entre vitesse et coût de propagation de l'information.

Le DNS est donc un système souple, extensible, et qui a fait ses preuves. De plus, c'est un protocole standard requis par les autorités du réseau *Internet*.

3.3 Constituants du DNS

Cette section expose les principes généraux de fonctionnement du DNS. Cette description n'est pas exhaustive, et ne remplace pas la lecture des *Requests For Comments* (voir [7][8][9][10]). Il s'agit là d'une introduction indispensable pour administrer un domaine.

Le DNS est constitué de trois composants :

3.3.1 L'espace de noms

La vision hiérarchique offerte à l'utilisateur (voir [1]) est une vision abstraite qui repose sur deux notions essentielles : les zones et les "Resource Records" (RR).

Zones et autorité

L'espace hiérarchisé est découpé en *zones* (ou *domaines*). Par exemple, dans la figure 3.1, la zone `jussieu` comprend l'ordinateur `shiva.jussieu.fr.` ainsi que les références aux zones des différents laboratoires. Au sommet de l'arbre figure la zone racine (*root*).

Il est important de comprendre qu'une zone ne correspond pas forcément à une branche de l'arbre, mais à un nœud et à ses descendants directs. Si un descendant est lui-même un nœud (et pas une feuille), ce peut être une nouvelle zone, comme à Jussieu. Par exemple, dans la figure 3.1, la zone `jussieu.fr` est limitée au rectangle délimité ; autrement dit, la zone `jussieu.fr` référence les zones `lptl.jussieu.fr` et `ccr.jussieu.fr` mais ne les contient pas.

Une entité, que nous appellerons dans un premier temps l'*autorité*, est responsable d'une ou de plusieurs zones. Ainsi, `shiva.jussieu.fr.` est responsable de la zone `jussieu` ainsi que de la totalité des zones des laboratoires. Si une machine au fin fond des Etats-Unis interroge `shiva.jussieu.fr.` pour connaître l'adresse de `bambi.lptl.jussieu.fr.`, la réponse sera garantie juste.

Si cette machine essaye d'obtenir l'adresse de `bambi.lptl.jussieu.fr.` à nouveau, elle trouvera l'information dans son cache. Cette information ne sera pas garantie. Le cache n'est pas l'*autorité*.

L'*autorité* se délègue : lorsque le DNS a été mis en place sur le réseau du Campus Jussieu, nous avons déposé une demande à l'Inria (qui est l'autorité du domaine `.fr`) pour avoir l'autorité sur `jussieu.fr`.

Les "Ressource Records"

Le contenu d'une zone est défini par des objets appelés "Resource Records" (RR). On peut voir un RR comme une ligne d'information élémentaire. Par exemple :

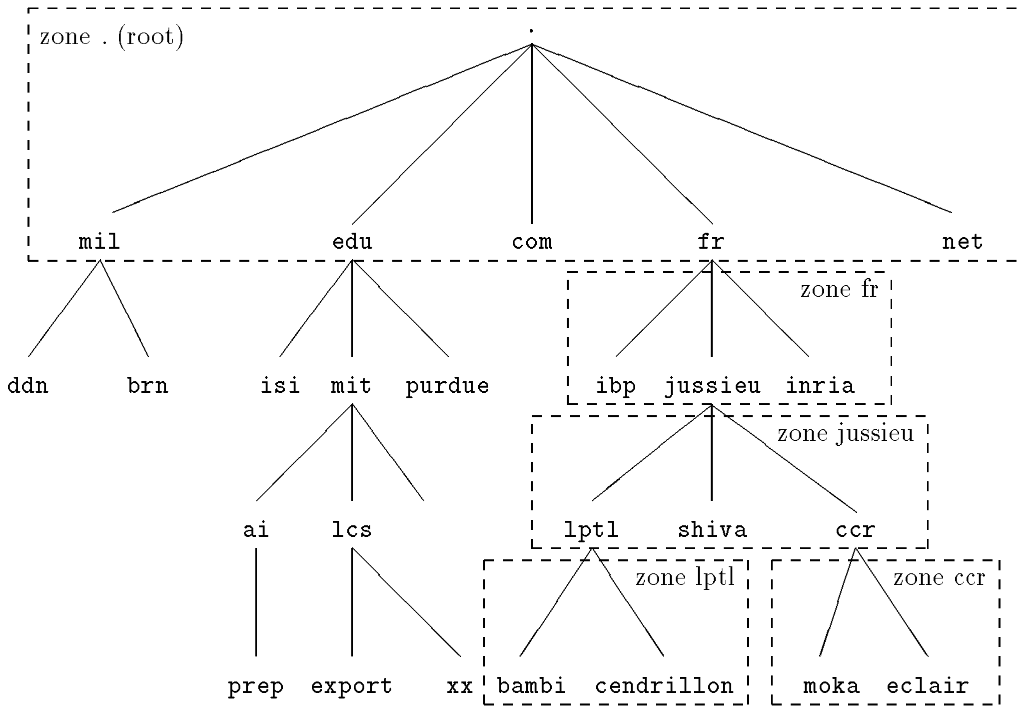

Figure 3.1: Découpage de la hiérarchie en zones

```
shiva.jussieu.fr.        172800  IN  A    134.157.0.129
bambi.lptl.jussieu.fr.   172800  IN  MX   100 shiva.jussieu.fr.
jussieu.fr.              172800  IN  NS   shiva.jussieu.fr.
```

Le premier RR est associé à `shiva.jussieu.fr.`, il peut rester au maximum 172800 secondes (c'est à dire deux jours) dans un cache au fin fond des Etats-Unis. Ce RR spécifie une classe `IN` (pour INternet). Une autre classe est définie, mais elle ne nous est pas utile. Le type (ici `A`) spécifie que le RR fournit l'adresse de l'objet (ici `shiva.jussieu.fr.`). Ce RR est logiquement dans la zone `jussieu.fr.`.

Le deuxième RR (associé à `bambi.lptl.jussieu.fr.`) indique que tout le courrier doit être transmis à `shiva.jussieu.fr.`. Voir le chapitre consacré à `sendmail` pour plus d'explications sur cette manière d'envoyer le courrier. Ce RR est logiquement dans la zone `lptl.jussieu.fr.`.

Le troisième RR (associé à `jussieu.fr.`, c'est à dire à une zone) spécifie qu'un serveur de noms réside sur `shiva.jussieu.fr.`. Ce RR est logiquement dans la zone `fr.`.

Heureusement, certains des champs inscrits dans l'exemple sont optionnels... Pour le détail des RR, de leurs composants et de leur syntaxe, le lecteur intéressé est renvoyé à [9].

3.3.2 Le serveur de noms

L'*autorité* dont nous avons parlé en 3.3.1 est implémentée sous la forme d'un "serveur de noms" (*name-server*). Un serveur de noms gère éventuellement une ou plusieurs zones. C'est lui qui maintient les RR et qui répond aux requêtes des clients.

Tous les serveurs de noms (au moins sur Unix) disposent d'un cache comme décrit en 3.2.3.

Le problème d'un serveur de nom ayant l'*autorité* (serveur *primaire*) pour une zone est que la machine qui l'héberge peut devenir indisponible, ou même ralentie. Pour éviter un blocage du système, l'*autorité* est distribuée à quelques autres serveurs. Sans avoir la possibilité de modifier les données, ces serveurs sont dits *secondaires*. Une zone doit être servie par des serveurs de noms suffisamment éloignés (géographiquement, électriquement, administrativement) pour qu'un dysfonctionnement local ne bloque pas tout le système.

Pour synthétiser les différences entre les deux types de serveurs (primaire et secondaire) :

- ils ont tous deux l'autorité pour une zone donnée ;
- du point de vue de l'extérieur (c'est-à-dire des clients du serveur de noms et des autres serveurs), ils sont rigoureusement équivalents ;
- l'administrateur édite les RR (i.e. les fichiers de configuration) sur le serveur primaire, ces RR étant automatiquement propagés vers les serveurs secondaires au bout d'un certain temps.

Un serveur de noms peut gérer plusieurs zones. Il peut être primaire pour certaines, secondaire pour d'autres. La configuration d'un serveur de noms est paramétrable en fonction des zones qu'il doit gérer, de l'activité prévue, et de la politique générale du site. Ceci sera étudié en détail dans 3.5.

3.3.3 Le "resolver"

Le "resolver" est le dernier constituant du DNS. C'est, sur Unix, une bibliothèque (fonction `gethostbyname` par exemple) qui, quand elle est incluse dans les programmes, fait appel aux serveurs de noms pour trouver l'information demandée.

Un "resolver" doit connaître au moins un serveur de noms pour débuter ses recherches. Par exemple, le serveur de noms utilisé par le "resolver" de `picsou.lptl.jussieu.fr.` est sur `cendrillon.lptl.jussieu.fr.`

Par exemple, supposons qu'un utilisateur sur la machine `picsou.lptl.jussieu.fr.` utilise la commande `rlogin` pour se connecter sur la machine `prep.ai.mit.edu`. Le programme `rlogin` contient le "resolver". Celui-ci va donc interroger le serveur de noms désigné (sur `cendrillon.lptl.jussieu.fr.`). Le serveur de noms, n'ayant pas `prep.ai.mit.edu.` dans ses zones ni dans son cache, va envoyer une requête à un des serveurs de la zone "`.`" (*root*) en demandant un serveur pour la zone `edu`. Ensuite, il s'adresse au serveur de cette zone `edu` pour avoir les adresses des serveurs de la zone `mit.edu`. Le processus continue ainsi jusqu'à obtention de l'information recherchée et transmission au "resolver" de `picsou.lptl.jussieu.fr.` qui a fait la demande.

Cet exemple correspond au cas d'un serveur de noms acceptant les requêtes *récursives*. Si ce n'est pas le cas, la requête est dite *itérative* et le "resolver" doit lui-même interroger les serveurs de noms un par un. Heureusement, l'implémentation du serveur de noms sur Unix autorise les requêtes récursives.

3.4 Conversion d'adresse IP en nom

Nous avons vu comment était réalisée la conversion de nom de machine en adresse IP. La conversion inverse est également utile.

Par exemple, lorsqu'un utilisateur sur la machne `a` se connecte sur la machine `b`, la machine `b` obtient l'adresse IP, c'est à dire l'adresse numérique, de `a`. Cette adresse est convertie en nom de machine pour :

- vérifier les autorisations de `/etc/hosts.equiv` ;
- vérifier les autorisations individuelles dans le fichier `.rhosts` de l'utilisateur ;
- enregistrer ce nom dans le fichier des utilisateurs connectés. Ce nom est visible lorsqu'on utilise la commande `who` par exemple.

Comment se passe cette conversion inverse ? Comment une machine fait pour reconnaître un nom lorsqu'elle ne dispose que de l'adresse IP ?

Pour résoudre ce problème, les concepteurs avaient d'abord envisagé un mécanisme : les requêtes inverses. Puis, ils se sont aperçus qu'il n'était point besoin de générer un cas particulier et ils ont mis en place le domaine `in-addr.arpa.`. Ce domaine (ou plutôt pseudo-domaine) contient des RR de type `PTR` correspondant aux adresses de machines. Ainsi, pour une machine d'adresse *aaa.bbb.ccc.ddd*, le nom du RR sera *ddd.ccc.bbb.aaa*`.in-addr.arpa.` (voir figure 3.2). Ce RR est en fait un *pointeur* vers les vrais RR de la machine.

Pour obtenir le nom d'une machine, il suffit donc au "resolver" d'inverser les quatre nombres constituant l'adresse IP, d'y accoler `in-addr.arpa` et d'envoyer une requête au serveur de noms.

Le réseau du Campus Jussieu dispose de la délégation pour la zone `157.134.in-addr.arpa.`.

3.5 Différents types de configurations du DNS

Les serveurs de noms peuvent être paramétrés pour une variété de configurations. Cette section récapitule les différents schémas disponibles.

Ces configurations peuvent être sélectionnées par modification (ou absence) du fichier `/etc/named.boot`[3].

3.5.1 Pas de serveur de noms !

La configuration de loin la plus simple à gérer, tout en disposant des facilités offertes par le DNS, est de ne pas avoir de serveur de noms du tout ! Dans ce cas, seul le "resolver" est activé (seul le fichier de configuration du "resolver" nécessite d'être présent). Le serveur de noms est alors sur une autre machine.

Cette configuration est de loin la plus courante, étant donné qu'un serveur de noms nécessite une administration relativement importante et génère du trafic réseau supplémentaire.

[3] `/usr/etc/named.boot` sur Silicon Graphic

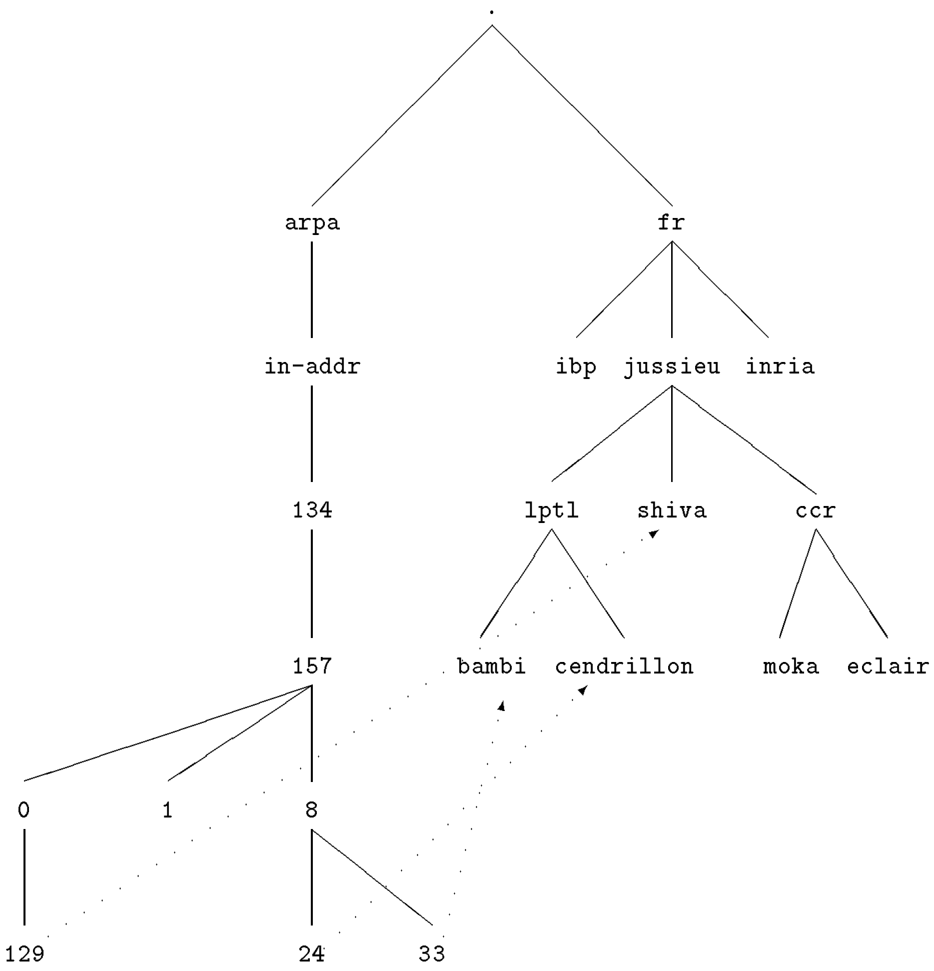

Figure 3.2: Le pseudo-domaine `in-addr.arpa`.

3.5.2 Gestion des zones

Comme indiqué en 3.3.2, un serveur peut gérer une ou plusieurs zones. Pour chaque zone gérée, il faut déterminer si le serveur doit être primaire ou secondaire. Seules les données du serveur primaire peuvent être modifiées. Les données du serveur secondaire sont mises à jour automatiquement à partir du serveur primaire lorsqu'il y a eu une modification.

Serveur primaire pour une zone

Déclarer qu'un serveur est primaire pour une zone revient à insérer, dans `/etc/named.boot`, une ligne comme :

```
primary     ccr.jussieu.fr        ccr
```

Il ne faut pas oublier le domaine pour la conversion inverse :

```
primary     1.157.134.in-addr.arpa   1
```

Les différents champs sont :

1. le mot-clef `primary` introduit une zone ;
2. le nom de cette zone est le deuxième champ ;
3. le dernier champ est le nom du fichier où se trouvent les RR de la zone (par exemple : `ccr` ou `1`).

Serveur secondaire pour une zone

Déclarer qu'un serveur est secondaire pour une zone revient à insérer une ligne comme :

```
secondary   ccr.jussieu.fr           134.157.0.129   ccr
secondary   1.157.134.in-addr.arpa   134.157.0.129   1
```

Les différents champs sont :

1. le mot-clef `secondary` introduit une zone ;
2. le nom de la zone est le deuxième champ ;
3. le troisième champ est l'adresse IP (ce ne peut être un nom) du serveur primaire ;
4. le dernier champ est le nom d'un fichier où le serveur place une copie de la zone. En temps normal, le serveur répond aux requêtes des clients en consultant son cache.

Le fichier n'est consulté que lorsque le serveur démarre et que le primaire ne répond pas. Il est initialisé et actualisé automatiquement, sans intervention manuelle. Il peut ne pas exister au moment où le serveur est lancé

Serveur cache seulement

Puisque tout serveur de noms inclut un cache, un serveur qui ne gère aucune zone (aucune ligne `primary` ou `secondary`) est un serveur *cache seulement*. C'est un cas particulier de configuration. La gestion des zones est réduite à 0 zone...

Le serveur n'a donc aucune *autorité*, il ne fait que répondre aux questions des "resolver".

3.5.3 Options de configuration

Nous décrivons ici les principales options de configuration qui affectent le comportement d'un serveur de noms. De même que précédemment, ces options sont activées par des lignes dans le fichier `/etc/named.boot`.

forwarders

L'option `forwarders` indique un certain nombre de serveurs de noms vers lesquels sont routées les requêtes non trouvées dans le cache. Le but de l'option `forwarders` est d'avoir un serveur général, et les autres serveurs utilisent l'option `forwarders` vers ce serveur général. Ainsi, le cache du serveur général s'enrichit de toutes les requêtes, et permet ainsi d'éviter des requêtes superflues vers le monde extérieur.

Lorsque le ou les serveurs généraux sont indisponibles, le serveur envoie alors la requête comme s'il n'y avait pas cette option.

slave

L'option `slave` est un cas à part. Elle nécessite l'option `forwarders`. La seule différence avec le cas ci-dessus est qu'en cas d'échec vers les serveurs généraux, la requête est considérée comme échouée.

Cette option permet aux réseaux sans "statut connecté" (c'est à dire sans les droits d'accès à tout le réseau *Internet*) d'accéder aux services du DNS.

Cette option est sans intérêt sur le Campus Jussieu.

3.6 Le DNS sur le réseau du Campus Jussieu

La mise en place du DNS sur le réseau du Campus Jussieu s'est faite progressivement dans les laboratoires.

3.6.1 Historique

Le DNS est installé depuis juillet 1990 sur le Campus Jussieu. L'installation s'est faite en plusieurs étapes :

1. installation d'un serveur de noms expérimental, sans "resolver", sur les machines `shiva.jussieu.fr.` et `cendrillon.lptl.jussieu.fr` dans une arborescence minimum servie par une machine du LPTL, jusqu'en février 1991 ;
2. déclaration du domaine `jussieu.fr.` dans les tables de `fr.` à l'Inria, avec trois serveurs reconnus (`shiva.jussieu.fr.`, `cendrillon.lptl.jussieu.fr.` et `layon.inria.fr.`) ;
3. mise à disposition opérationnelle pour les utilisateurs du LPTL à partir du 13 février 1991 ;
4. mise à disposition opérationnelle pour les utilisateurs du CCR sur la machine `ccr-np.ccr.jussieu.fr.` à partir du 25 mars 1991 ;
5. réunion d'information aux administrateurs système le 10 avril 1991 et début de la configuration définitive ;
6. depuis, l'immense majorité des laboratoires utilise le DNS.

Cette section a pour but d'expliquer comment configurer votre laboratoire dans le DNS.

3.6.2 La politique de Jussieu

Chaque laboratoire (ou plus généralement *service*) connecté au réseau du Campus Jussieu doit avoir un nom de domaine. Ce nom de domaine doit suivre les règles de l'*Internet*, et être accepté par l'équipe réseau du Campus. En particulier, tout nom de domaine composé de deux lettres ou homonyme d'un *top-level domain* (`fr`, `edu`, etc.) sera rejeté.

Les noms de domaines actuellement enregistrés[4] sont listés en annexe.

Le but à atteindre est que chaque laboratoire dispose de son propre serveur de noms gérant sa zone de manière à simplifier l'administration et la mise à jour. En attendant, toutes les zones non gérées par les laboratoires sont gérées par le serveur de nom principal de Jussieu, à savoir `shiva.jussieu.fr.`.

Afin de faciliter l'administration du réseau du Campus et pour que notre site soit conforme aux règles de l'*Internet* et de Renater, certaines règles élémentaires sont à respecter.

Lorsque vous allez mettre en place votre serveur gérant votre zone primaire, vous devrez prévenir l'équipe réseau du Campus Jussieu et déclarer comme serveurs secondaires les machines suivantes :

- `shiva.jussieu.fr.` (134.157.0.129)
- `cendrillon.lptl.jussieu.fr.` (134.157.8.24)
- `layon.inria.fr.` (192.93.2.4)

[4] Au 1er avril 1993.

L'équipe réseau vérifiera que vos zones sont correctes et les enregistrera dans la zone `jussieu.fr.`.

Afin d'éviter trop de trafic sortant vers l'extérieur, vos serveurs de noms doivent être configurés avec l'option `forwarders` vers `shiva.jussieu.fr.` et `cendrillon.lptl.jussieu.fr.`. Ne mettez pas trop de serveurs de noms en route : il vaut mieux un cache bien rempli que beaucoup de petits caches forcés d'obtenir les informations à l'extérieur...

Vos zones, pour des raisons de sécurité, ne doivent pas inclure de RR de types `HINFO` et `WKS`.

N'oubliez pas que vous devez gérer la zone de votre laboratoire, mais également la zone dans `in-addr.arpa.` correspondant à votre numéro de *subnet* alloué.

3.7 Le DNS et `sendmail`

Les règles de fonctionnement du réseau Jussieu imposent une interaction entre le DNS et la messagerie.

Les RR de type `MX` spécifient une machine préférée pour la réception du courrier. Par exemple, le RR de l'exemple en 3.3.1 spécifie que tout le courrier adressé à `bambi.lptl.jussieu.fr.` doit être envoyé à `shiva.jussieu.fr.`.

Les règles de Jussieu précisent que toutes les machines doivent avoir des RR de type `MX` comme celui de `bambi.lptl.jussieu.fr.` afin d'assurer un traitement cohérent et homogène. Si une machine au fin fond des Etats-Unis veut envoyer un courrier à `bambi.lptl.jussieu.fr.`, ce courrier sera envoyé à `shiva.jussieu.fr.` qui, après traitement, l'enverra à `bambi.lptl.jussieu.fr.`.

Si cela fonctionne bien avec une machine du fin fond des Etats-Unis, cela pose en revanche un problème à l'intérieur de Jussieu. si la machine `picsou.lptl.jussieu.fr.` essaye d'envoyer un courrier à `bambi.-lptl.jussieu.fr.`, le courrier sera envoyé par erreur à `shiva.jussieu.fr.`, ce qui provoque au mieux un aller et retour inutile, et au pire un va-et-vient sans interruption entre `shiva.jussieu.fr.` et la machine de courrier du laboratoire concerné.

La solution au problème est d'utiliser une version de `sendmail` qui utilise le serveur de noms (pour obtenir les adresses IP) mais pas les RR de type `MX`. Une telle version est généralement disponible, ou sinon en *ftp anonyme* sur `ftp.jussieu.fr.`.

3.8 Installation

Cette section ne se substitue pas au manuel de votre ordinateur (voir [14]), mais donne quelques petits trucs utiles qui ont déjà servi...

3.8.1 Obtention des informations de base

La première chose à faire est de récuperer par *ftp anonyme* sur `ftp.jussieu.fr.` les fichiers se trouvant dans le répertoire `pub/named/config`.

Profitez de l'occasion pour regarder s'il n'y a pas des nouveautés intéressantes pour vous.

3.8.2 Configurer un serveur de noms

Fichier `/etc/named.boot`

Il faut choisir un répertoire (`/usr/local/named` par exemple) pour placer les informations du serveur de noms. Ce répertoire sera indiqué par la ligne comme :

```
directory /usr/local/named
```

Après avoir placé le fichier `root.cache` (obtenu en 3.8.1) dans le répertoire choisi, mettez ensuite la ligne dans votre `/etc/named.boot` :

```
cache . root.cache
```

N'oubliez pas de mettre la ligne :

```
forwarders 134.157.0.129 134.157.8.24
```

Vous pouvez ensuite déclarer vos zones avec `primary` comme décrit en 3.5.2.

Fichier de configuration des zones

Le mieux pour cela est de reprendre un fichier existant. Il est vraisemblable qu'une zone existe déjà sur `shiva.jussieu.fr.` pour votre laboratoire. Demandez à l'équipe réseau le fichier correspondant à votre zone et complétez-le.

La première erreur la plus répandue dans la configuration d'une zone est l'oubli des points finaux, surtout dans la zone x`.157.134.in-addr.arpa.`...

La deuxième erreur la plus répandue est l'oubli de la modification du *numéro de version* dans le champ `SOA` pour *toute* modification dans votre zone. Cette erreur est perverse et ne se manifeste que pour les "resolver" s'adressant aux serveurs secondaires. Vous ne la verrez donc pas, mais les autres sites en seront génés.

Une autre erreur fréquente consiste à oublier les espaces autour des parenthèses dans les RR `SOA` et `WKS`.

3.8.3 Configurer un "resolver"

La configuration du "resolver" se fait simplement à l'aide du fichier `/etc/resolv.conf`[5]. Ce fichier doit contenir au moins deux lignes :

`domain` *labo*`.jussieu.fr.`
`nameserver` *l'adresse-IP-de-votre-name-server*

Que votre serveur de noms soit sur une autre machine (par exemple `shiva.jussieu.fr.`) ou sur votre machine, vous devez avoir ce fichier. Vous pouvez installer plusieurs serveurs de noms (plusieurs lignes *nameserver*) afin d'assurer un minimum de fiabilité.

3.8.4 Tester un serveur de noms

Il vaut mieux tester votre serveur de noms avant de le configurer de manière opérationnelle. Pour cela, il existe un outil nommé `nslookup`[6].

Lorsque vous démarrez `nslookup`, vous interrogez le serveur de noms de votre machine locale ou celui indiqué dans `/etc/resolv.conf`.

Essayez une requête telle que *machine.labo*`.jussieu.fr.`. Si cela marche, vous pouvez essayer une requête en dehors de votre laboratoire. Essayez alors `moka.ccr.jussieu.fr.`. Si cela continue à marcher, essayez d'enlever le point final. Si ça marche, vous pouvez essayer enfin une machine au fin fond des Etats-Unis...

Si une des requêtes ne marche pas, essayez de trouver les serveurs ayant l'*autorité* pour le domaine correspondant. Faites `set q=ns`, puis essayez par exemple `ccr.jussieu.fr.`.

Si ces requêtes ne marchent pas, vérifiez que le serveur de noms fonctionne toujours, qu'il n'a pas envoyé de message d'erreur à la console. Si tout est normal, mettez `nslookup` en mode *test* : `set debug`. Réiterez alors.

Lorsque ces requêtes donnent satisfaction, vous pouvez passer aux conversions inverses. Faites `set q=ptr` et essayez *y.x*`.157.134.in-addr.arpa.`. Si ça ne marche pas, il doit y avoir quelque chose d'incorrect dans votre configuration[7]...

3.8.5 Cas spéciaux

Si le serveur de noms et son implémentation sous Unix ne varient pas, il n'en va pas de même de tout ce qu'il y a autour : mise en œuvre du "resolver", interaction avec `sendmail`, signaux de contrôle, etc.

Cette partie est destinée à exposer les petites particularités de chaque système. Si le vôtre n'est pas dans les cas ci-dessous, faites le nous savoir et nous l'incluerons dans les prochaines versions de ce document.

[5] `/usr/etc/resolv.conf` sur Silicon Graphic.
[6] Sauf sur les machines IBM sous AIX
[7] Ce genre d'indication est utile, n'est-ce pas ?

Sun

Les ordinateurs Sun ne tiennent en standard pas compte du DNS. Sous les versions SunOS 4.x, on peut résoudre le problème.

SunOS 4.0 : Sous SunOS 4.0, la solution au problème est de chercher la librairie `libc.so`... adaptée à votre machine en *ftp anonyme* sur `ftp.jussieu.fr.` dans le répertoire `pub/named/sunos-4.0`.

Une fois cette librairie installée (et prise en compte avec `ldconfig`), configurez votre `/etc/resolv.conf`.

Ce patch fonctionne avec et sans les NIS (anciennement pages jaunes). Attention : le programme `rcp` continue à fonctionner avec le fichier `/etc/hosts`. Contactez les auteurs si vous êtes dans ce cas.

SunOS 4.1 : Les choses sont plus compliquées car il faut tenir compte de deux cas :

1. vous utilisez les NIS (anciennement *pages jaunes*). Dans ce cas, mettez la variable `B` à la valeur `-b` dans le fichier `Makefile` des NIS, et reconfigurez vos pages jaunes avec `ypinit -m`. Sur chaque serveur (NIS), mettez un fichier `/etc/resolv.conf` et lancez `ypserv` avec l'option `-d`.
2. vous n'utilisez pas les NIS. Dans ce cas, il vous suffit d'utiliser le *ftp anonyme* sur `shiva.jussieu.fr.` dans le répertoire `pub/named/sunos-4.1` (ou le répertoire correspondant à votre version de système) pour trouver la `libc.so`... adaptée à votre machine. N'oubliez pas de faire `ldconfig` après avoir installé la librairie.

   D'autre part, le programme `rcp` est compilé statiquement et tient donc toujours compte du fichier `/etc/hosts`. Une version modifiée est disponible en *ftp anonyme*.

Hewlett-Packard

Sur les ordinateurs Hewlett-Packard, `sendmail` tient obligatoirement compte des RR de type `MX`. Vous devez donc installer la version de `sendmail` adaptée à votre machine (HP 9000/300 ou 400, ou HP 9000/700 sous HP-UX 8.0 et ultérieur) disponible en *ftp anonyme*, dans le répertoire `pub/sendmail/binaires`.

Silicon Graphic

Sur les ordinateurs Silicon Graphic, `sendmail` tient obligatoirement compte des RR de type `MX`. Vous devez donc installer la version de `sendmail` disponible en *ftp anonyme*, dans le répertoire `pub/sendmail/binaires`.

Digital Equipement

Les ordinateurs Digital Equipement sous Ultrix disposent d'un fichier de configuration nommé `/etc/svc.conf`. La ligne `hosts=` doit indiquer le mot-clef `bind`. Par exemple :

```
hosts=bind,local
```

Encore - Gould

Sur les matériels Gould de Encore, il faut modifier le fichier `/etc/hostdb_method` pour indiquer qu'il faut consulter le DNS.

3.9 Trucs et astuces

Quelques petits conseils très pratiques vous seront sans doute utiles :

Tout d'abord, la mise en place du DNS rend obsolète le fichier `/etc/hosts`. Une bonne pratique consiste cependant à continuer de le mettre à jour pour les machines de votre laboratoire. En effet :

- lors du démarrage du système Unix, certains utilitaires utilisent le fichier `/etc/hosts`, par exemple lorsque le serveur de noms n'a pas encore démarré ;
- le format de ce fichier est plus lisible que les fichiers de configuration de zones. Il est donc plus facile de consulter manuellement ce fichier pour certaines opérations ;
- si, pour une raison ou pour une autre, vous suspendez temporairement l'utilisation du DNS, ce fichier doit être à jour.

Dans ce fichier `/etc/hosts`, placez le nom pleinement qualifié en premier. Par exemple :

```
134.157.8.33     bambi.lptl.jussieu.fr     bambi
```

D'autre part, le démarrage d'un système Unix est une période très instable. Pour éviter des mauvaises surprises, placez des adresses numériques lors des appels à `ifconfig` ou à `route` dans les fichiers de boot `/etc/rc` ou `/etc/rc.local`.

N'oubliez pas de qualifier (c'est à dire de mettre les noms complets, sans point final) dans le fichier `/etc/exports` si vous utilisez NFS, et dans les fichiers `/etc/hosts.equiv` et `.rhosts` chez chaque utilisateur.

Enfin, pendant la période d'installation, n'hésitez pas à redémarrer votre ordinateur. Beaucoup de problèmes dus à des états transitoires se sont réglés de cette façon. Lorsque vous aurez achevé de mettre en place votre DNS, lorsque vous aurez tout terminé et que tout marchera, rebootez une dernière fois. Comme cela, vous serez sûr que tout fonctionne normalement, même après une coupure d'électricité !

Chapitre 4

La messagerie du réseau du Campus Jussieu

4.1 Introduction

Ce document a pour but de répondre aux questions suivantes :

- comment sont organisées les messageries *Internet* et *Earn/Bitnet* ;
- comment est organisée la messagerie sur le Campus Jussieu ;
- adressage et sendmail ;
- le fichier de configuraton de `sendmail` (`sendmail.cf`) ;
- que faire pour s'insérer dans la structure de messagerie Jussieu.

Il accompagne la version courante du *sendmail pour tous* et doit permettre son installation aisée sur toute machine appartenant au réseau du Campus Jussieu.

Le DNS est supposé opérationnel dans tout ce qui suit.

4.2 Les messageries Internet et Earn/Bitnet

4.2.1 Internet

Internet est le réseau des machines fonctionnant avec les protocoles TCP/IP. Il offre trois importantes fonctionalités :

1. le terminal virtuel (telnet) ;

2. le transfert de fichiers (ftp) ;
3. le transport du courrier (smtp).

Théoriquement, toute machine connectée à ce réseau peut ouvrir une connexion vers toute autre pour assurer le service correspondant. En pratique, il y a des restrictions (principalement pour des questions de sécurité) à cette puissante interconnexion.

La connexion de bout en bout est un élément très important de tous les services offerts par le réseau *Internet*.

Toutefois, pour ce qui est du courrier, ce principe est moins intéressant : il pose des problèmes dans la mesure où toute machine doit être au faîte des subtilités de la messagerie.

Pour éviter cela, le Campus Jussieu a dédié une machine (appelée `mailhost.jussieu.fr`) à l'acheminement des courriers avec l'extérieur et l'intérieur du Campus. Cette machine dispose de la connaissance du routage des messages et décharge ainsi les machines des laboratoires de problèmes tels que :

- mise à jour de la liste des laboratoires pour le courrier intérieur ;
- connaissance des passerelles vers les autres réseaux de messagerie ;
- disponibilité permanente ;
- etc.

Les RR de type MX ne sont pas utilisés à l'intérieur du Campus Jussieu. Toutefois, `mailhost.jussieu.fr` devant s'adresser à toutes les autres machines du réseau *Internet*, cette machine utilise les RR de type MX pour les connexions extérieures.

4.2.2 *Bitnet*

Bitnet (qui porte également le nom *Earn* en Europe) est un réseau d'origine IBM ne regroupant que des centres de recherche. Il offre principalement les deux fonctionalités suivantes :

1. le transport du courrier ;
2. le transfert de fichiers.

Toute machine connectée au réseau connaît la liste complète des autres machines du réseau ainsi que, pour chaque machine, le nom du nœud suivant du chemin permettant de la joindre.

Un courrier passe donc de machine en machine (*Store and Forward*) selon un chemin que l'on espère optimum pour parvenir à la machine destinatrice. Sur une machine Unix, le protocole de transport NJE[1] est assuré par le logiciel UREP, qui permet l'échange de courriers sur lignes IP grâce au protocole NJE/IP.

[1] Protocole propriétaire IBM.

4.2.3 X400

La connexion au réseau de messagerie X400 est envisagée à court terme à Jussieu.

4.3 La messagerie à Jussieu

4.3.1 Principes

La messagerie à Jussieu est basée sur trois principes : arborescence, simplicité et fiabilité.

L'arborescence (voir figure 4.1) permet à chaque niveau d'avoir l'information nécessaire et suffisante pour router un courrier sans erreur. Chaque nœud de l'arbre est une machine gérant le courrier entre les nœuds inférieurs qui appartiennent à son domaine et le nœud supérieur qui est son relai vers le reste du monde. C'est ce qu'on a appelé un *mailhost*.

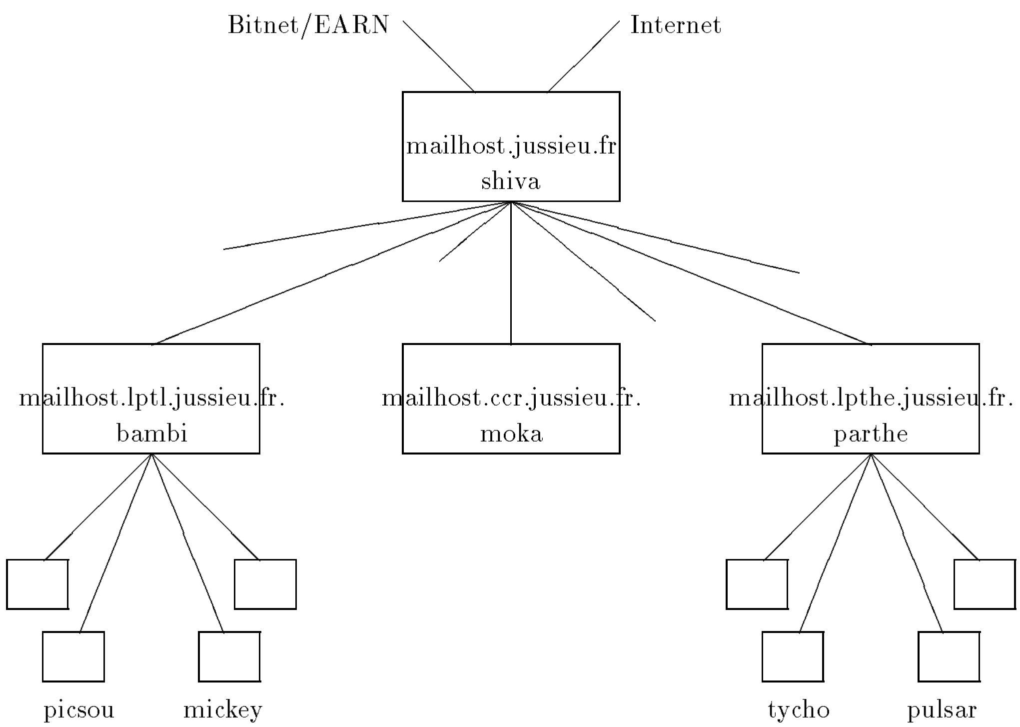

Figure 4.1: Organisation hiérarchisée de la messagerie

La machine `shiva.jussieu.fr` a la connaissance du domaine `.jussieu.fr`. Cette machine est donc le *mailhost* de Jussieu. De même, cette machine est le nœud *Bitnet* `FRUNIP62`.

4.3.2 Le mailhost de jussieu

Cette machine (*mailhost.jussieu.fr*) effectue :

- la réception des courriers venant de l'extérieur (de tout le réseau *Internet* ou du nœud *Bitnet* voisin) ;
- le routage vers les laboratoires ;
- le routage du courrier inter-laboratoires ;
- l'envoi via *Internet* ou *Bitnet* selon l'adresse du destinataire (à la machine destinatrice ou au nœud *Bitnet* voisin).

Il n'y a aucun compte utilisateur, donc aucune boîte aux lettres, sur cette machine.

La machine dediée à ces tâches s'appelle `mailhost.jussieu.fr`[2]. Une procédure de secours est prévue en cas de panne.

4.3.3 Mailhost de laboratoire

Chaque laboratoire gérant son courrier a un *mailhost*. Cette machine a pour tâche :

- de gérer les courriers pour le laboratoire : c'est à dire gérer l'espace des noms (elle doit connaître d'une manière ou d'une autre les noms de login et/ou les noms "en clair", éviter les homonymes...) et la localisation des boîtes aux lettres ;
- de distribuer le courrier local ;
- de renvoyer au *mailhost* de Jussieu ce qui ne concerne pas le laboratoire.

Elle applique la politique de distribution de courrier du laboratoire.

L'appellation `mailhost.`*laboratoire*`.jussieu.fr` permet, grâce aux serveurs de noms, de changer de *mailhost* facilement.

4.4 Adressage et `sendmail`

L'adressage est basé sur la *Request For Comment* 822 ([6]). Une adresse est constituée d'une partie *local*, d'un `@` et d'une partie *domaine* :

local`@`*domaine*

[2] Son vrai nom est actuellement `shiva.jussieu.fr`.

La partie *local* est utilisée par le site destinataire et ne doit pas être interprétée par les sites intermédiaires. Dans le cas où une passerelle de messagerie doit être explicitement désignée (réseaux non implicitement interconnectés) les `@` de la partie locale sont réécrits en `%` pour empêcher leur interprétation.

La partie *domaine* (voir [9]) est prise dans un espace de noms hiérarchisé et structuré en domaines et sous-domaines séparés par des points (`.`). Le *domaine* s'interprète de droite à gauche ; l'élément le plus à droite étant le *top-level domain*.

Un *top-level domain* est soit le nom normalisé de pays (par exemple : `fr`, `uk`, `fi`...) soit, pour les Etats-Unis uniquement, un type d'organisme (par exemple : `edu`, `com`, `mil`...)

Les adresses ne correspondant pas à cette norme sont traitées de la même manière grâce aux *pseudo top-level domains*. Un *pseudo top-level domain* permet de traiter temporairement les autres adresses au format décrit dans [6] (par exemple : `uucp`, `bitnet`...).

La migration d'une partie des réseaux de messagerie *NJE* et *SMTP* vers la messagerie normalisée X400 de l'ISO est prévisible à moyen terme. Cette partie sera donc développée dans les versions suivantes de ce document.

4.4.1 L'adressage à Jussieu

Les adresses à Jussieu peuvent prendre deux formes selon le réseau de messagerie utilisé.

Internet

L'adresse *Internet* comporte :

1. un champ *domaine* :

   L'idéal est de ne plus avoir que des adresses en 3 sous-champs :

   `....@`*laboratoire*`.jussieu.fr`

   Il faut donc que tout courrier ait des champs `From:` et/ou `To:` sous cette forme quand ils concernent Jussieu. Cela nécessite qu'un laboratoire :

   - recevant une telle adresse, sache délivrer le courrier ;
   - émettant un courrier, débarrasse le champ `From:` de toute référence à une machine.

   La disparition du nom de machine dans l'adresse de courrier est très intéressante car :

   - les adresses communiquées aux correspondants sont simples et font abstraction des détails du réseau du laboratoire ;
   - si le laboratoire change de machine, l'adresse reste la même ;
   - si un utilisateur change de machine dans son laboratoire, son adresse reste également la même.

2. un champ *local* : le *nom-de-login* pourra être remplacé par le *nom-en-clair* :

   Prénom.Nom`@`*laboratoire*`.jussieu.fr`

Pour cela, le fichier des *alias* peut être utilisé (attention aux homonymes). Le fichier des *alias* (habituellement `/usr/lib/aliases` ou `/etc/aliases` permet d'associer un nom quelconque à un *nom de login*).

Il peut cependant y avoir des cas où un quatrième sous-champ[3] est nécessaire :

- il n'y a pas *un* espace de noms unique pour tout le laboratoire : le *mailhost* ne sait pas faire parvenir un courrier à la boîte à lettres destinatrice à partir des seules informations que sont l'adresse `...@`*laboratoire*`.jussieu.fr` *et* son fichier des *alias* ;
- le laboratoire gère des sous-laboratoires ou équipes (au sens courrier du terme).

Un laboratoire devrait pouvoir éviter ce quatrième champ en recourant aux *alias* sur le *mailhost*. Cette possibilité de conserver le quatrième champ a été laissée bien qu'à l'encontre du principe de simplicité. Le `sendmail.cf` fourni n'a pas été testé dans ce cas de figure.

Bitnet

L'adresse *Bitnet* comporte :

1. un champ *domaine* limité à 2 sous-champs `...@frunip62.bitnet` ;
2. un champ *local* identique à ce qui a été décrit ci-dessus.

Le *mailhost* de Jussieu fait suivre les adresses *x*`@frunip62.bitnet` vers le CCR.

Toutefois, il est possible de se faire enregistrer comme "utilisateur Bitnet" afin que le *mailhost* de Jussieu fasse suivre les courriers vers le *mailhost* du laboratoire. Cet enregistrement est, bien entendu, indispensable pour envoyer du courrier d'un laboratoire vers le monde Bitnet.

Le problème des homonymes se pose avec acuité. Il n'est pas possible d'enregistrer deux utilisateurs différents avec le même nom. Par exemple, `toto@labo1.jussieu.fr` et `toto@labo2.jussieu.fr` ne peuvent pas tous les deux être enregistrés en tant que `toto@frunip62.bitnet`. Ces problèmes sont traités au cas par cas.

Utilisation des services communs

Les utilisateurs qui ne peuvent ou ne veulent pas recevoir et/ou envoyer des courriers à partir de leurs laboratoires pourront toujours utiliser `moka.ccr.jussieu.fr` (qui est le *mailhost* du CCR) comme support courrier.

Leur adresse *Internet* est alors

`...@ccr.jussieu.fr`

[3] C'est à dire des adresses de la forme `....@`*machine.laboratoire*`.jussieu.fr`.

et l'adresse *Bitnet* est toujours

```
...@frunip62.bitnet
```

4.4.2 Sendmail

Un courrier est composé de trois parties :

1. l'enveloppe (informations échangées entre mailers) ;
2. l'entête (informations standards lisibles par l'utilisateur ajoutées et/ou modifiées par les sendmails) ;
3. le corps (transmis tel quel).

`Sendmail` est *le* logiciel assurant le routage du courrier sur une machine Unix. Ce logiciel est complètement[4] paramètrable à l'aide du fichier `sendmail.cf`.

`Sendmail` sait :

- détecter les erreurs d'adressage (à son niveau) ;
- livrer un courrier dans une boîte à lettres locale ;
- envoyer un courrier à une autre machine selon
  - l'adresse du destinataire,
  - la connaissance qu'il a de la politique de routage,
  - les messageries auxquelles il a directement accès

  en utilisant le protocole de transport de courrier adéquat (smtp, uucp, nje, x400...) ;
- réécrire les champs `From:` et `To:` en fonction des réseaux origine et destination.

`Sendmail` est appelé :

- par la commande `mail` ou tout autre logiciel UA (*User Agent*) : `mailx`, `elm`... ;
- par smtp ou autre MTA (*Message Transfert Agent*) : uucp, nje, x400... ;
- en direct (pour ceux qui utilisent `sendmail` comme une commande...) ;
- par le démon `sendmail` lui-même qui va périodiquement vérifier si la file d'attente des messages bloqués (en général à cause de l'indisponibilité de la machine destinatrice) n'est pas vide et, dans ce cas, tenter un nouvel envoi.

`Sendmail` prend ses messages :

[4] Mais pas aisément !

- dans la *mail-queue* ;
- selon le protocole smtp ;
- sur l'entrée standard de la commande.

`Sendmail` route :

- en constituant (ou modifiant) une enveloppe ;
- en complétant/modifiant l'entête ;
- en appelant le mailer approprié.

`Sendmail` connaît les *mailers* suivants :

- error : défini d'office, il permet, lorsqu'une erreur est détectée de renvoyer le courrier à son expéditeur si possible et, dans certains cas, au *Postmaster* de la machine (toute machine gérant du courrier doit obligatoirement déclarer un *Postmaster*, usuellement par l'intermédiaire du fichier des *alias*, auquel on peut d'ailleurs envoyer directement du courrier pour demander un renseignement du type "nom d'un correspondant local"). Le *Postmaster* doit garder un œil sur les erreurs locales car elles peuvent mettre en évidence un disfonctionnement de `sendmail`. Les erreurs détectées dépendent de la connaissance que le `sendmail` de la machine sur lequel il s'exécute a du routage. Ainsi par exemple, `mailhost.jussieu.fr.` peut détecter un envoi à un laboratoire inexistant mais pas à un utilisateur inconnu dans un laboratoire.
- local : assure la distribution des courriers dans les boîtes à lettres de cette machine.
- smtp : transfert de courriers sur protocole IP (*Internet*).
- nje : transfert de courriers utilisé pour *Bitnet* (utilise maintenant IP comme support à Jussieu).
- x400 : à venir.

La figure 4.2[5] montre l'intégration et la position de `sendmail` dans l'architecture de courrier d'une machine Unix.

4.5 Les fichiers de configuration de `sendmail`

Il y a donc principalement trois types de `sendmail.cf` pour Jussieu. Écrits à partir de la même base, ils assurent cohérence et unicité de comportement du plus simple (`sendmail.cf` d'une feuille) au plus compliqué (`sendmail.cf` de `mailhost.jussieu.fr.`).

[5] Cette figure est inspirée de [14]

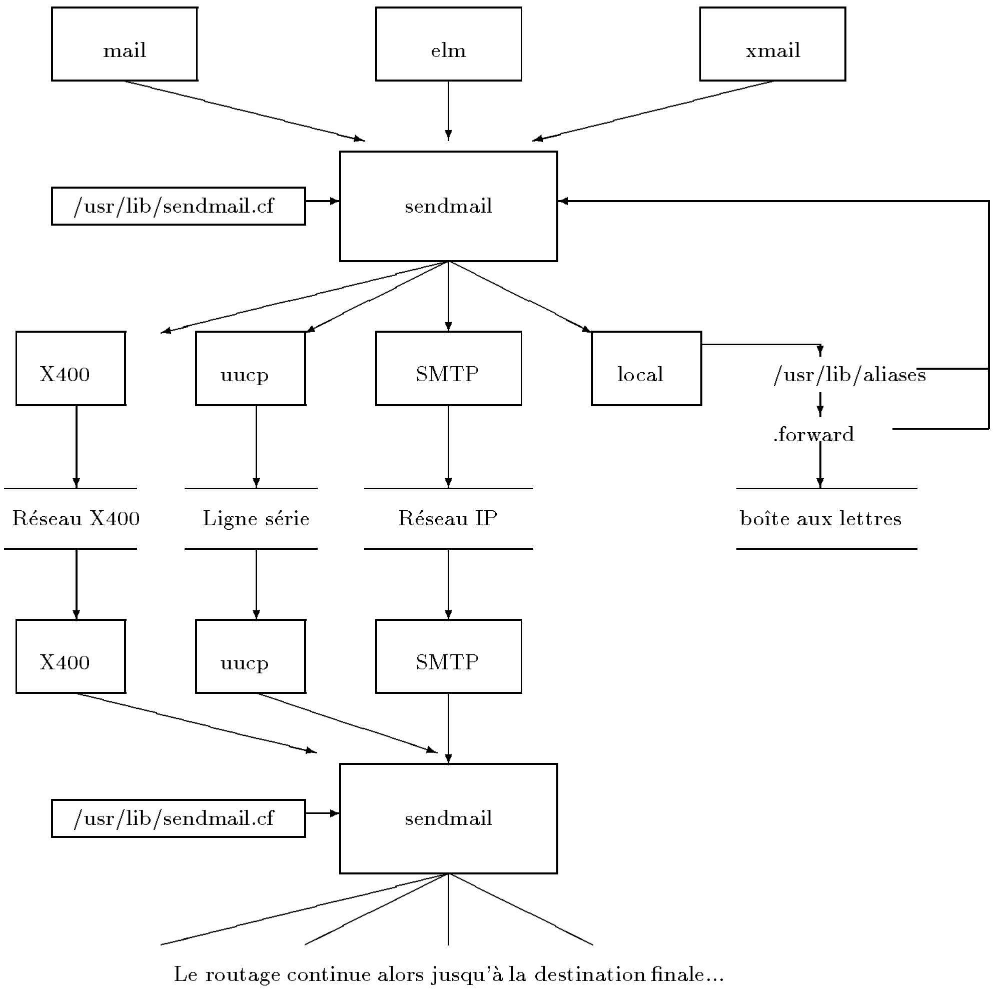

Figure 4.2: Routage du courrier avec `sendmail`

4.5.1 Le `sendmail.cf` de `mailhost.jussieu.fr.`

`Mailhost.jussieu.fr` gère **`.jussieu.fr`**. Tout courrier entrant ou sortant de Jussieu via *Internet* ou *Bitnet* passe par **`mailhost.jussieu.fr`**. Cette machine connaît tous les laboratoires et leurs *mailhosts*. Le routage interne à Jussieu se fait donc sur le champ *laboratoire* de

`...@`*laboratoire*`.jussieu.fr`

Il y a donc interface unique avec chaque laboratoire.

La machine **`moka.ccr.jussieu.fr`** (qui est le *mailhost* du CCR) est le serveur de courrier des utilisateurs des laboratoires non équipés.

4.5.2 Les `sendmail.cf` des *mailhosts* des laboratoires

Outre une meilleure sécurité, l'interface unique entre **`mailhost.jussieu.fr.`** et les laboratoires permet :

- la simplicité et la fiabilité des **`sendmail.cf`** (donc une configuration automatique possible grâce à la *normalisation*) ;
- la connaissance par **`sendmail`** des seules informations nécessaires et suffisantes pour assurer sa fonction ;
- la maintenance assurée par une équipe unique.

Un *mailhost* connaît son laboratoire. C'est à ce niveau que se décide la politique de routage des courriers du laboratoire. Le *mailhost* doit savoir s'il est *la* machine ayant les boîtes aux lettres, si **`/usr/spool/mail`** est partagé (NFS) ou s'il doit faire suivre (et vers quelle machine du laboratoire).

Pour la sécurité, il serait souhaitable que ces *mailhosts* n'acceptent des connexions smtp que de **`mailhost.-jussieu.fr.`** et des machines du laboratoire.

4.5.3 Les `sendmail.cf` des feuilles

Selon la politique de routage retenue par le laboratoire, la machine doit savoir si elle stocke les courriers locaux au site ou si elle doit faire suivre vers le *mailhost* qui en sait plus quant à la localisation des boîtes aux lettres des utilisateurs du laboratoire.

Cette dernière solution, centralisatrice, permet une grande souplesse de gestion des boîtes aux lettres :

- les courriers deviennent indépendants des machines ;
- les utilisateurs se choisissent une boîte préférentielle ;
- l'expéditeur est sûr que le courrier arrivera dans la boîte que consulte habituellement son correspondant et non sur une machine ou il se connecte rarement.

Cette solution est très intéressante, malgré une connexion smtp peut être inutile, eu égard à la simplicité induite.

Pour des raisons de sécurité, il serait souhaitable que ces machines n'acceptent des connexions smtp que de leur *mailhost*.

4.6 Installation

Chaque laboratoire voulant s'insérer dans cette stucture doit choisir une machine pour servir de *mailhost* (indépendamment de la machine servant de serveur de noms). Le laboratoire doit également choisir une politique de distribution du courrier.

4.6.1 Choix du laboratoire

Le choix de la politique de distribution du courrier est du ressort du laboratoire. Il y a plusieurs cas possibles.

En réception :

1. Il n'y a qu'une seule machine entreposant le courrier :
   $\Longrightarrow$ c'est alors le *mailhost*.
2. Le spool, c'est à dire l'espace disque sur lequel sont stockées les boîtes aux lettres (`/usr/spool/mail` sur BSD, `/usr/mail` sur System V) est partagé par NFS :
   - par toutes les machines :
     $\Longrightarrow$ le *mailhost* écrit directement dans les boîtes aux lettres ;
   - par une partie des machines du laboratoire :
     $\Longrightarrow$ le *mailhost* garde les courriers concernant ces machines (on lui fournira la liste à la configuration) et reroute vers les autres en fonction d'informations supplémentaires (quatrième champ, alias...).
3. Le courrier est reçu par toute machine de laboratoire :
   $\Longrightarrow$ le *mailhost* garde les courriers le concernant et reroute les autres courriers vers les autres machines en fonction d'informations supplémentaires (quatrième champ, alias...).

En émission :

1. Le champ `From:` doit être du type `...@`*laboratoire*`.jussieu.fr` :
   - pour toute les machines :
     $\Longrightarrow$ le *mailhost* "normalise" ;
   - pour une partie des machines du laboratoire :
     $\Longrightarrow$ le *mailhost* "normalise" pour les courriers émanant de ces machines (on lui fournira la liste à la configuration) et ne touche pas au champ `From:` des courriers émanant des autres (idem 2)).

2. Le champ `From:` doit laisser passer le quatrième champ :
   $\Longrightarrow$ le *mailhost* ne fait aucun traitement sur ce champ.

4.6.2 Mode d'emploi

Ce qu'il faut faire pour configurer le courrier pour *mailhost* de laboratoire :

1. récupérer l'exemple de configuration (`exemple.config`) et le configurateur (`configurateur`) par *ftp anonyme*[6] sur `ftp.jussieu.fr`, dans le répertoire `pub/sendmail/config`
2. modifier l'exemple pour l'adapter au *mailhost* du laboratoire et à la politique de distribution du courrier choisie
3. rendre le configurateur exécutable :

   `chmod +x configurateur`
4. exécuter le configurateur

   `configurateur mon-exemple > sendmail.cf.new`
5. si vous êtes courageux, testez la configuration :

   `sendmail -bt -Csendmail.cf.new`
6. terminer le démon `sendmail` s'il existe

   `kill -15` *pid-de-sendmail*
7. mettre le `sendmail.cf` courant en réserve et copier le nouveau

   `cd /usr/lib` ou `cd /etc`
   `mv sendmail.cf sendmail.cf.old`
   `mv $HOME/sendmail.cf.new sendmail.cf`
8. lancer le démon `sendmail` avec les paramètres habituels (récupérés dans le fichier `/etc/rc.local`), par exemple :

   `/usr/lib/sendmail -bd -q1h`
9. tester grandeur nature.

le `sendmail.cf` des feuilles pouvant être configuré indifféremment sur la machine destinatrice ou sur une machine tierce, pour chaque machine du laboratoire, modifier l'exemple pour l'adapter à la machine cible et recommencer à l'exécution du configurateur) sans oublier le transport du `sendmail.cf.new` obtenu sur cette machine cible !

Tout au long de ces étapes, vous pouvez rencontrer des difficultés ou... des erreurs de configuration. Les auteurs sont à votre disposition pour toute assistance et/ou maintenance.

Un grand nombre de machines sont déjà configurées et fonctionnent de manière satisfaisante. Il n'a pas été possible de tester tous les cas de figure sur tout matériel et/ou sous tout les Unix existants. Il faut que vous reportiez les problèmes et/ou informations auprès des auteurs qui en tiendront compte dans les futures versions.

[6] Vous pouvez par la même occasion regarder s'il y a des choses dans `/pub/sendmail` concernant votre configuration.

4.7 Conclusion

Cette version de configurateur sera suivie de nouvelles versions pour les habituelles correction d'erreurs, améliorations diverses, ajouts de cas de figure, etc...

Afin de vous faciliter les mises à jour, faites-nous connaître les modifications typiques de votre laboratoire. Elles seront alors intégrées et vous n'aurez plus à modifier le configurateur.

Bon courage.

Annexe A

Les zones de Jussieu

Domaine	Signification
	Services communs
`ccr`	Centre de Calcul Recherche
`reseau`	Matériels de gestion du réseau
	Laboratoires
`acg`	Analyse Complexe et Géometrie
`aero`	Aéronomie Jussieu
`aerov`	Aéronomie Verrières
`analyse`	Equipe d'Analyse
`ann`	Analyse Numérique
`aomc`	Accoustique et Optique de la Matière Condensée
`artemis`	Didactique des Disciplines
`b3e`	Biomathématiques, Biostatistique, Bioinformatique et Epidémiologie (CHU St Antoine)
`biogeodis`	Biogéochimie, Géologie Appliquée et "International geosphere and biosphere program"
`biomath`	Biomathématiques (CHU La Pitié-Salpétriere)
`cmt`	Chimie des Matériaux de Transition
`cot`	Chimie Organique Théorique
`dga`	Département de Géologie Appliquée
`dim`	Dynamique des Interactions Moléculaires
`drp`	Département de Recherches Physiques
`elchp7`	Electrochimie Moléculaire de Paris VII
`gcr`	Institut Henri Poincaré
`geois`	Géochimie des Isotopes Stables
`gps`	Groupe de Physique des Solides

Domaine	Signification
groupes	Théorie des Groupes
ipgp	Institut de Physique du Globe de Paris
isup	Institut de Statistiques de Paris VI
itodys	Institut de Topologie et Dynamique des Systèmes
ladl	Automatisme Documentaire et Linguistique
lbhp	Biorhéologie et Hydrodynamique Physico-Chimique
lcpp6	Chimie-physique Paris VI
lcs	Cristallochimie du Solide
ldimo	Dispositifs Infrarouges et Micro-ondes
ldma	Dynamique Moléculaire et Atomique
let	Echanges Thermiques
lgs	Géologie Structurale
lif	Informatique Fondamentale
linguist	Unité de Formation et Recherches Linguistiques
lmcp	Minéralogie Cristallographie et Pétrologie
lmd	Météorologie Dynamique
lmf	Mathématique Fondamentale
lmm	Modélisation en Mécanique
lodyc	Océanographie Dynamique et Climatologie
logique	Logique Mathématique
lpan	Physique Atomique et Nucléaire
lpcm	Physique et Chimie Marine
lpdp	Physiologie du Développement des Plantes
lpma	Physique Moléculaire et Atmosphérique
lpmm	Pétrologie Mineralogie Métallogenie
lpthe	Physique Théorique des Hautes Energies
lptl	Physique Théorique des Liquides
lrs	Réactivités de Surface et Structures
lsm	Synthèse Macromoléculaire
magsur	Magnétisme des Surfaces
mathp6	Mathématiques Paris VI
mathp7	Mathématiques Paris VII
phymath	Physique Théorique et Mathématique
plecnrs	Physique des Liquides et Electrochimie
pmc	Physique des Milieux Condensés
ppce	Propriétes Physico-chimiques des Electrolytes
proba	Probabilités

Domaine	Signification
radium	Physique Chimie de l'Institut Curie
robo	Robotique de Paris
snv	Institut des Neurosciences
spectro	Spectroscopie Hertzienne de l'ENS
spmol	Spectrochimie Moléculaire
sri	Structure et Réactivité aux Interfaces
ufrphyp7	Physique Paris VII
urbb	Unité de Recherches Biomathématiques et Biostatistiques
	Autres Services
cicrp	Centre Interuniversitaire De Calcul de la Région Parisienne (enseignement)
sgp7	Secrétariat Général Paris VII
sigp6	Service Informatique de Gestion Paris VI
sigu7	Service de Gestion de Paris VII

Annexe B

Exemples de configuration de serveur de noms

Ce chapitre montre un exemple de configuration disponible en *ftp anonyme* sur `ftp.jussieu.fr.` dans le répertoire `pub/named/config`.

B.1 /etc/named.boot

```
;
; Fichier de configuration pour un serveur de noms de laboratoire
;
; Historique
;   91/04/10 : pda : creation d'une configuration exemple
;

;
; Nos reseaux preferentiels
;
sortlist        134.157.0.0 132.227.0.0

;
; Repertoire utilise par ce serveur de noms
;
directory       /usr/local/named

;
; Nom du fichier contenant le cache
;
;cache          domaine                 fichier
;-------------- ----------------------- ----------------------------------------
cache           .                       root.cache

;
; Domaines geres par ce serveur de noms
```

```
;
;primary        domaine                 fichier source
;-------------- ----------------------- ---------------------------------------
primary         LABO.jussieu.fr         LABO

;
; Ce serveur de noms envoie toutes ses requetes a shiva.jussieu.fr ou
; a cendrillon.lptl.jussieu.fr.
;
forwarders      134.157.0.129           134.157.8.24

;
; Pseudo-domaines pour la conversion adresse -> nom host
;
;primary        domaine                         fichier source
;-------------- ------------------------------ --------------------------------
primary         0.0.127.in-addr.arpa            localhost.rev
primary         NUMERO.157.134.in-addr.arpa     LABO.rev
```

B.2 exemple.zone

```
;
; Zone LABO.jussieu.fr.
;
; Historique
;   91/04/10 : jt/pda : creation d'une zone exemple
;

;
; Designation de l'autorite. Debut d'une zone.
;
;               SOA     machine                  personne en charge
;               ------- ------------------------ ----------------------
@       IN      SOA     SERVEUR.LABO.jussieu.fr. MOI.LABO.jussieu.fr. (
                                9201281          ; Version
                                21600            ; Refresh (6h)
                                3600             ; Retry   (1h)
                                3600000          ; Expire
                                172800 )         ; Minimum (2j)
        IN      NS      SERVEUR.LABO.jussieu.fr.
        IN      NS      shiva.jussieu.fr.
        IN      NS      cendrillon.lptl.jussieu.fr.
        IN      NS      soleil.uvsq.fr.
        IN      NS      layon.inria.fr.

;
; Designation des noeuds du domaine lptl.jussieu.fr.
;

MACHINE1        IN      A       134.157.NUMERO.1
                IN      MX      100     shiva.jussieu.fr.
```

```
MACHINE2        IN      A       134.157.NUMERO.2
                IN      MX      100     shiva.jussieu.fr.

;
; Definition des aliases du laboratoire
;

mailhost        IN      CNAME   MACHINE1
```

B.3 exemple.rev

```
;
; Zone NUMERO.157.134.in-addr.arpa.
;
; Historique
;   91/04/10 : jt/pda : creation d'une zone exemple
;

;
; Designation de l'autorite. Debut d'une zone.
;
;               SOA     machine                  personne en charge
;               ------- ------------------------ ----------------------
@       IN      SOA     SERVEUR.LABO.jussieu.fr. MOI.LABO.jussieu.fr. (
                                9201141          ; Version
                                21600            ; Refresh (6h)
                                3600             ; Retry   (1h)
                                3600000          ; Expire
                                172800 )         ; Minimum (2j)
        IN      NS      SERVEUR.LABO.jussieu.fr.
        IN      NS      shiva.jussieu.fr.
        IN      NS      cendrillon.lptl.jussieu.fr.
        IN      NS      soleil.uvsq.fr.
        IN      NS      layon.inria.fr.

0       IN      PTRLABO-net.jussieu.fr.

;
; LABO
;
1       IN      PTR      MACHINE1.LABO.jussieu.fr.
2       IN      PTR      MACHINE2.LABO.jussieu.fr.

;
; Machines de service
;
126     IN      PTR      r-LABO.reseau.jussieu.fr.
```

B.4 localhost.rev

```
;
; Zone 0.0.127.in-addr.arpa.
;
; Historique
;   91/04/10 : pda : creation d'une zone exemple
;


;
; Designation de l'autorite. Debut d'une zone.
;
;               SOA     machine                  personne en charge
;               ------- ------------------------ ---------------------
@       IN      SOA     SERVEUR.LABO.jussieu.fr. MOI.LABO.jussieu.fr. (
                                9011201          ; Version
                                21600            ; Refresh (6h)
                                3600             ; Retry   (1h)
                                3600000          ; Expire
                                172800 )         ; Minimum (2j)
        IN      NS      SERVEUR.LABO.jussieu.fr.

1       IN      PTR     localhost.
```

B.5 /etc/resolv.conf

Le fichier `/etc/resolv.conf` représente la configuration du "resolver".

```
;
; resolv.conf pour l'acces a un serveur de noms
;
; Historique
;   91/04/10 : pda : creation d'un exemple
;


;
; Le nom de domaine
;
domain LABO.jussieu.fr.
;
; Le serveur de noms local s'il est configure. Omettre cette ligne si
; vous ne desirez pas mettre en place un serveur de noms.
;
nameserver 134.157.NUMERO.1
;
; Les autres serveurs du campus en cas de defaillance du serveur local, ou
; lorsque le serveur de noms local n'est pas configure.
; Attention : sur la plupart des implementations, on est limite a
; 3 lignes "nameserver" au total.
;
nameserver 134.157.0.129
nameserver 134.157.8.24
```

B.6 `root.cache`

Le fichier `root.cache` donne la liste des serveurs de la zone "." (*root*).

```
;
; Hints for root nameservers
;   04/10/91 : jt   : mise a jour
;
.                       999999  IN      NS      ns.nic.ddn.mil.
                        999999  IN      NS      kava.nisc.sri.com.
                        999999  IN      NS      aos.brl.mil.
                        999999  IN      NS      c.nyser.net.
                        999999  IN      NS      terp.umd.edu.
                        999999  IN      NS      ns.nasa.gov.
                        999999  IN      NS      nic.nordu.net.
;
ns.nic.ddn.mil.         999999  IN      A       192.112.36.4
kava.nisc.sri.com.      999999  IN      A       192.33.33.24
aos.brl.mil.            999999  IN      A       192.5.25.82
c.nyser.net.            999999  IN      A       192.33.4.12
terp.umd.edu.           999999  IN      A       128.8.10.90
ns.nasa.gov.            999999  IN      A       128.102.16.10
ns.nasa.gov.            999999  IN      A       192.52.195.10
nic.nordu.net.          999999  IN      A       192.36.148.17
```

Annexe C

Le fichier de configuration de la messagerie

C.1 exemple.config

Le fichier `exemple.config` est le fichier attendu par le configurateur pour générer un `sendmail.cf` prêt à être utilisé sur votre machine.

```
#
# Configuration du sendmail des laboratoires de jussieu.fr
#
# Auteurs :
#   Jacky Thibault (jt@jussieu.fr)
#   Pierre David (pda@jussieu.fr)
#
# Historique
#   91/04/03 : pda/jt : conception
#   91/11/28 : jt/pda : disparition des cas speciaux CCR et LPTHE
#   92/07/17 : jt/pda : simplification
#

#
# Changer les valeurs des variables ci-dessous en fonction de votre site.
# Attention : ne pas laisser d'espaces de part et d'autre du signe "="
# dans les affectations.
#

###############################################################################
# Parametres du site
###############################################################################

#
# Host
#
# Mettez ici le nom de la machine sans le nom de domaine
# Exemple : "bambi" pour "bambi.lptl.jussieu.fr".
```

```
#
Host="bambi"

#
# Labo
#
# Mettez ici le nom de votre laboratoire.
# Ca doit correspondre a la partie "laboratoire" du nom du domaine.
# Exemple : "lptl" pour "lptl.jussieu.fr"
#
Labo="lptl"

#
# Mailhost
#
# Indiquez ici si ce sendmail est destine a la machine servant de
# mailhost pour le laboratoire.
#
# Valeurs possibles :
#       o
#       n
#
Mailhost="o"

###############################################################################
# Variables pour le MAILHOST
# NE LES MODIFIER QUE POUR LA CONSTITUTION D'UN sendmail.cf D'UN mailhost
###############################################################################

#
# SpoolCommun
#
# Si "o", tout le courrier a destination du laboratoire (ou du groupe de
# machines defini par SpoolMachines) doit etre garde sur ce mailhost.
# Si "n", le courrier doit etre individuellement envoye a la machine du
# laboratoire.
#
# Valeurs possibles :
#       o
#       n
#
SpoolCommun="o"

#
# SpoolMachines
#
# Ensemble des machines pour lesquelles le courrier est garde sur le
# mailhost (soit parce que le mailhost est la machine ou tout le monde
# lit son courrier, soit parce que les autres machines montent le spool/mail
# par NFS).
# A initialiser seulement si SpoolCommun="o"
#
# Valeurs possible :
#       chaine vide si l'ensemble = tout le laboratoire
```

```
#       liste de machines separees par des espaces (incluant le mailhost)
#
# Exemple : SpoolMachines="bambi mickey jumbo picsou cendrillon"
#
SpoolMachines=""

#
# FromLabo
#
# Si "o", le nom du host est retire du champ "From: " pour ne laisser que
# le nom du laboratoire (ex : lptl.jussieu.fr au lieu de bambi.lptl.jussieu.fr)
# pour toutes les machines definies dans "FromLaboMachines"
# Si "n", le nom du host est laisse dans le champ "From: " pour tout le monde.
#
# Valeurs possibles :
#       o
#       n
#
FromLabo="o"

#
# FromLaboMachines
#
# Ensemble des machines pour lesquelles le mailhost doit retirer le host
# du champ "From: ".
# A initialiser seulement si FROMLABO="o"
#
# Valeurs possibles :
#       chaine vide si l'ensemble = tout le laboratoire
#       liste de machines separees par des espaces (incluant le mailhost)
#
# Exemple : FromLaboMachines="bambi mickey jumbo picsou cendrillon"
#
FromLaboMachines=""

################################################################################
# Variables pour les feuilles
# NE LES MODIFIER QUE POUR LA CONSTITUTION D'UN sendmail.cf D'UNE feuille
################################################################################

#
# ToutEnvoyer
#
# Si "o", le courrier local (ainsi que tous les autres courriers) doit etre
# envoye sur le mailhost et pas entrepose sur la feuille.
# Si "n", le courrier local doit rester sur cette machine.
#
# Attention ! La valeur mise ici doit etre homogene avec la politique
# adoptee dans votre laboratoire, definie sur le mailhost :
# si SpoolCommun="o" et (SpoolMachines="" ou la feuille est dans SpoolMachines)
#   alors ToutEnvoyer="o"
#   sinon ToutEnvoyer="n"
# fin si
#
```

```
# Valeurs possibles :
#       o
#       n
#
ToutEnvoyer="o"

###############################################################################
# Parametres du systeme
###############################################################################

#
# Pour determiner tous ces parametres, il faut localiser votre fichier
# sendmail.cf existant :
#       /etc/sendmail.cf                (Sun, IBM)
#       /usr/lib/sendmail.cf            (autres machines, y compris HP)
#       /etc/newconfig/sendmail.cf      (HP ne tournant pas deja sendmail)
#

#
# Aliases
#
# Pour determiner la valeur :
#       Utilisez la commande "grep '^OA' sendmail.cf"
#
# Valeurs possibles :
#       /usr/lib/aliases                (toutes machines sauf Sun)
#       /etc/aliases                    (Sun, IBM)
#
Aliases=/usr/lib/aliases

#
# SendmailSt
#
# Pour determiner la valeur :
#       Utilisez la commande "grep '^OS' sendmail.cf"
#
# Valeurs possibles :
#       /usr/lib/sendmail.st            (toutes machines sauf Sun)
#       /etc/sendmail.st                (Sun, IBM)
#
SendmailSt=/usr/lib/sendmail.st
```

Bibliographie

[1] David P., Thibault J. *L'intégration du réseau de Jussieu dans le réseau* Internet. Disponible sur demande auprès des auteurs.

[2] Comer D. *Internetworking with TCP/IP.* Second edition. Prentice Hall 1991.

[3] Nemeth E., Snyder G., Seebass S. *Unix System Administration Handbook.* Prentice Hall 1989.

[4] Tanenbaum A. *Computer Networks.* Second edition. Prentice Hall 1989.

[5] Postel J.B. *Simple Mail Transfer Protocol* Request For Comments 821, 1982

[6] Crocker D.H. *Standard for the Format of ARPA Internet Text Messages* Request For Comments 822, 1982

[7] Stahl M. *Domain Administrators Guide* Request For Comments 1032, 1987

[8] Lottor M. *Domain Administrators Operations Guide* Request For Comments 1033, 1987

[9] Mockapetris P. *Domain Names - Concepts and Facilities* Request For Comments 1034, 1987

[10] Mockapetris P. *Domain Names - Implementation and Specification* Request For Comments 1035, 1987

[11] Mockapetris P. *DNS Encoding of Network Names and Other Types* Request For Comments 1101, 1989

[12] Allman E. *Sendmail – An Internetwork Mail Router* Unix System Manager's Manual, BSD 4.2 (distribué avec le source de `sendmail`).

[13] Allman E. *Sendmail – Installation and Operation Guide* Unix System Manager's Manual, BSD 4.2 (distribué avec le source de `sendmail`).

[14] Hewlett-Packard *Installing and Administering ARPA services*, Chapter 4 Manuels de HP-UX 7.0, Septembre 1989