Une architecture de réseau sécurisée avec filtrages

UREC/CNRS - LAAS/CNRS

Mai 2000

L'internet a changé...

- Réseau académique → LE réseau (universel)

  Nombre de machines connectées en France :

1990	3 000
1994	7 000
1997	30 000
Fév 2000	1 264 000

- De plus en plus de problèmes de sécurité :
  Nombre mensuel d'incidents traités par le CERT Renater :
  97 : **10** - 98 : **20** 99 : **100** - 01/2000 : **190** - 06/2000 : **280**

- Nos réseaux et nos accès à Renater
  conçus en 94-95 pour un Internet académique « familial »

Rester totalement ouvert : inconscience

Les systèmes n'ont pas changé...

- Défauts de sécurité dans certains protocoles,
- Windows 9x/NT n'est pas mieux qu'Unix (pour la sécurité)
- Tous les systèmes ont des vulnérabilités (`securityfocus.com` 1999) :

Windows NT4SP4	29
Solaris 7	27
RedHat Linux 6.1	20

- Ils sont toujours livrés ouverts par défaut
  - Avec des services réseaux inutiles mais activés
  - Les administrateurs doivent faire le ménage
- De plus en plus de stations dans les labos et sur les campus...

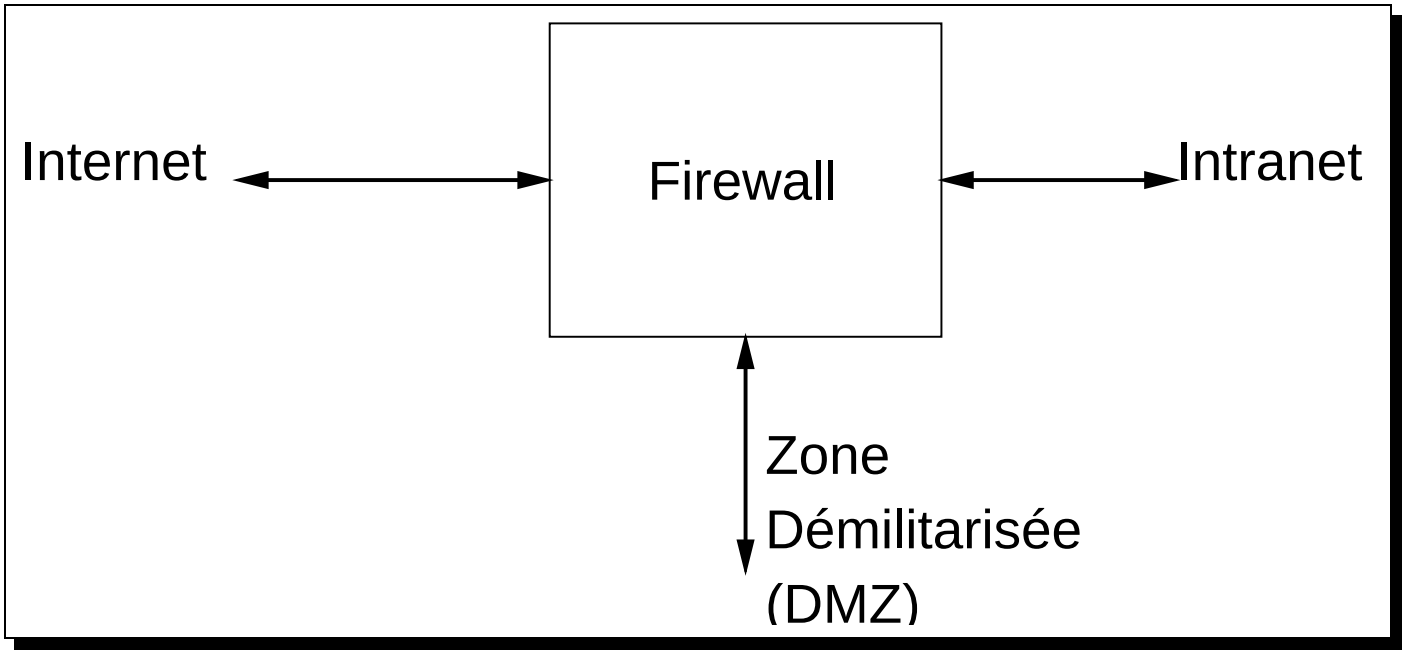
On ne peut pas maintenir la totalité de son parc sans trou de sécurité

Schéma classique d'attaque sur Internet

1. Scan du réseau pour découvrir les stations
2. Vérification des versions des démons et des services actifs
3. Attaque des versions avec trous de sécurité
4. Passage en mode administrateur et :
   - création de nouveaux utilisateurs
   - modification d'exécutables $\rightarrow$ portes dérobées
   - installation d'un *sniffer* : récupération des mots de passe
   - installation d'outils de déni de service distribués

Protection : bloquer ou limiter la portée de 2 et 3

Les Gardes-barrière (Firewalls)



Niveau application (relais applicatifs/proxys)

Niveau réseau (filtrage IP, VPs,...)

Niveau physique (VLans,...)

Architecture d'un site : principes

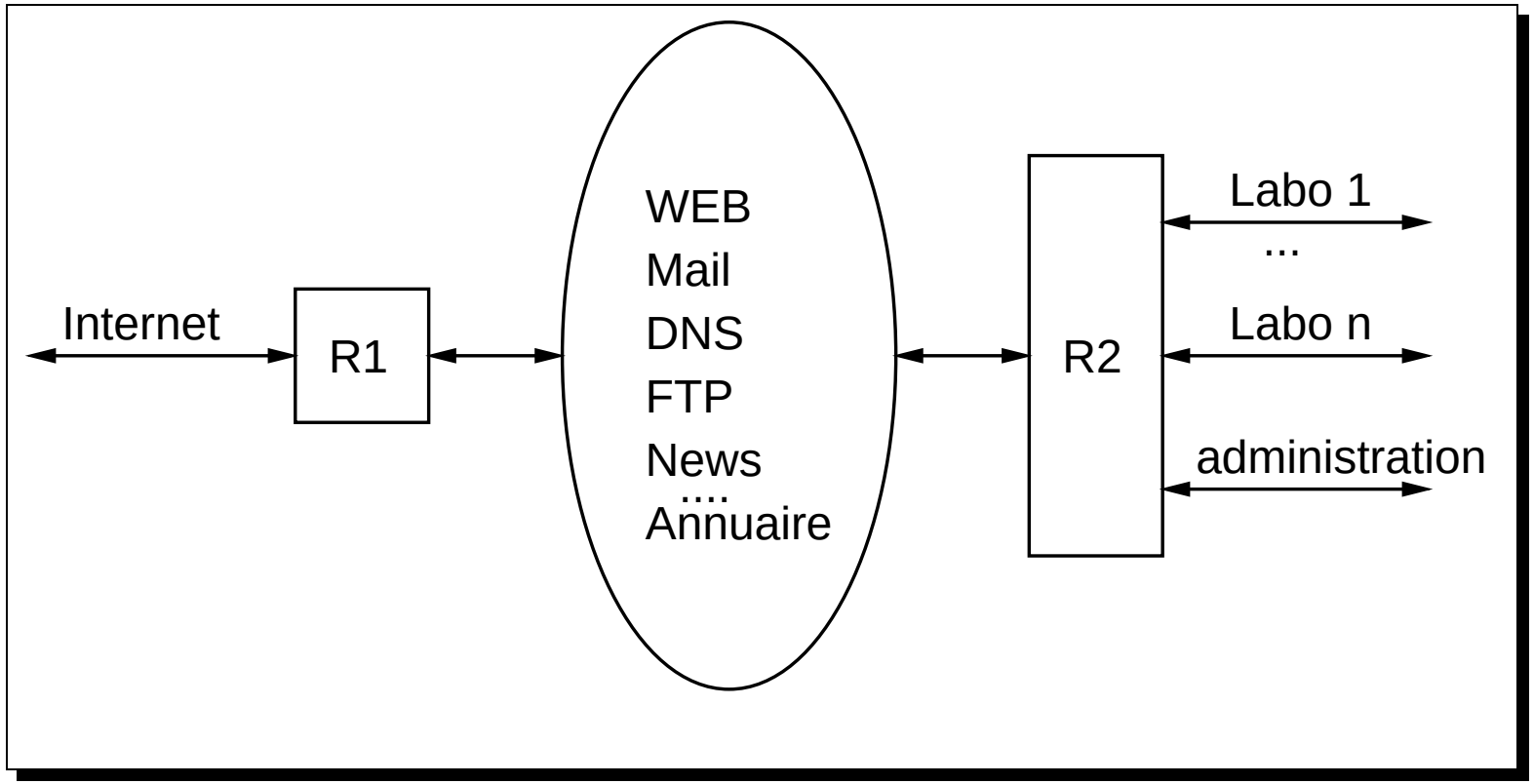
- Segmentation du réseau avec des routeurs
  - Entrée du site : zone semi-ouverte
    → serveurs réseau
  - Un segment (ou VLAN) par laboratoire / service...
- Flux des applications : controlés par filtrage
  - définir une politique
  - Où mettre les filtres ?
    → en entrée du site
    → entre segments (laboratoire/salles de cours/gestion)

Politique de filtrage

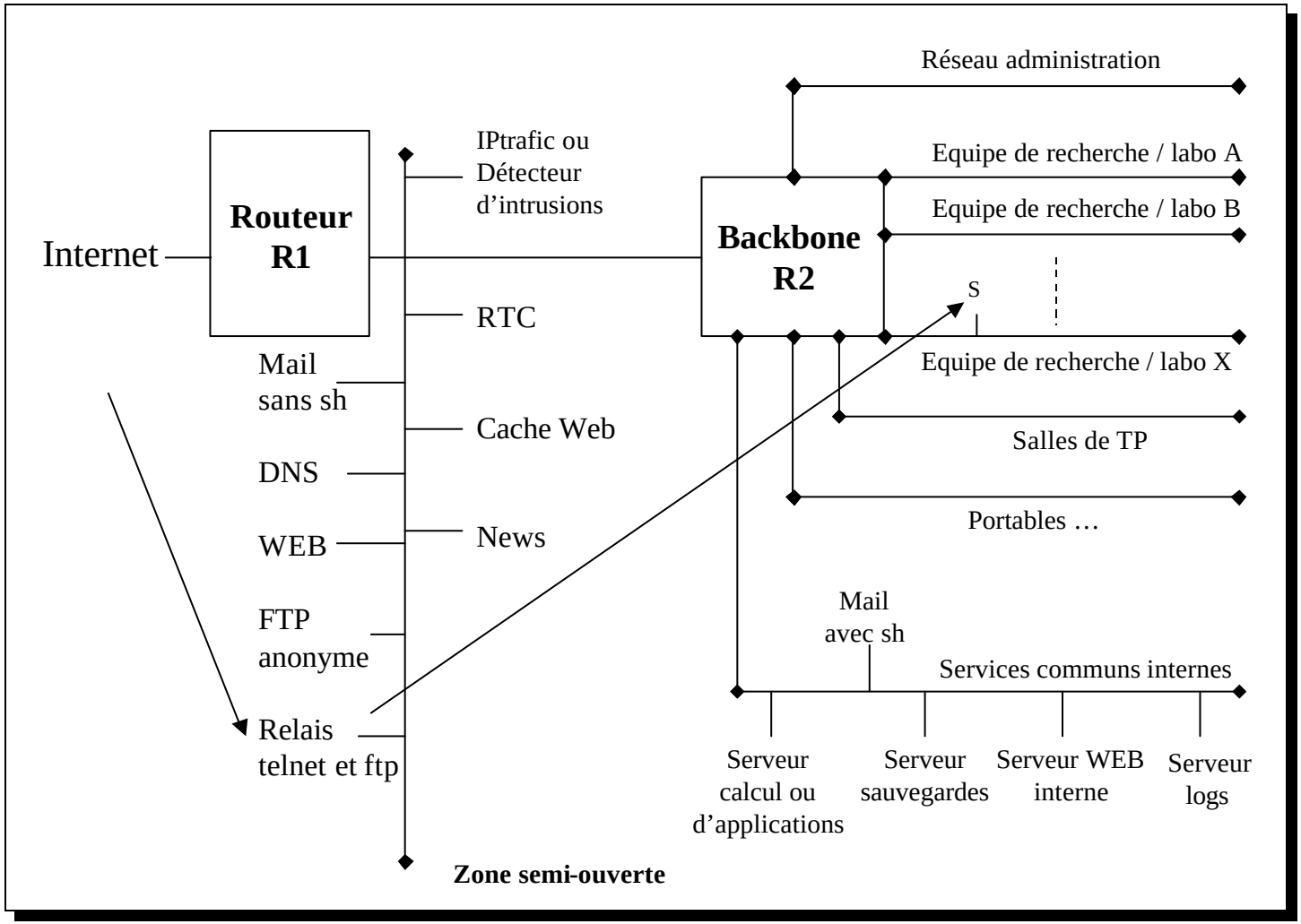
- **dans le sens sortant :** peu de limitations, toutes les stations peuvent être clientes (sauf exceptions : salles de formation...).
- **dans le sens entrant :**
  - Tout interdire par défaut,
  - Limiter l'accès aux services connus et bien administrés.
- **à l'intérieur :** segmenter les différentes communautés si nécessaire.

$\rightarrow$ séparation des machines qui offrent des services sur l'internet du reste du réseau.

Architecture : schéma simplifié



Architecture : schéma détaillé



Services dans la zone semi-ouverte

Web, FTP, DNS, mail (SMTP), News, base de données publiques, accès RTC,

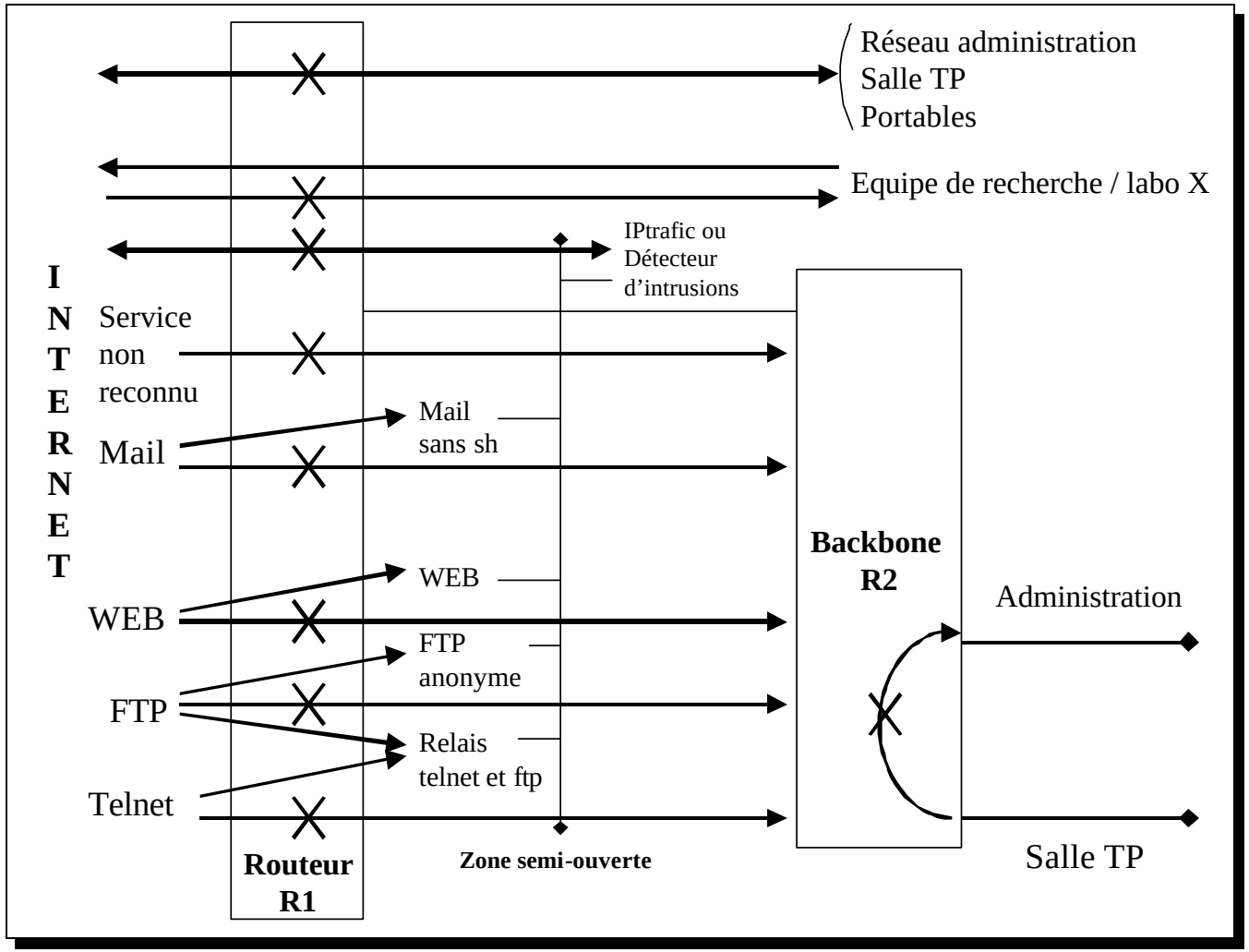
Machines dédiées à ces fonctions :
pas de comptes utilisateurs ni d'applications inutiles

Traces du traffic par `tcp_wrappers` dirigées vers une machine interne

Mail : serveur SMTP redirige les messages vers les BALs à l'intérieur.

Option : relais applicatifs telnet, POP/Imap,...

Architecture : filtres



Filtrage IP

- Sur R1 :

  Dans le sens entrant : tout interdire sauf services de la zone semi-ouverte autorise quelques services (SSH) vers la zone interne.

  Cacher des sous-réseaux à protéger complètement de l'internet

- Sur R2 :

  R2 peut avoir des fonctionnalités plus primaires.

  Rôle : protection des entités internes entre elles :

  restreindre les accès à l'administration

  Possibilité de faire du NAT sur un réseau

Bénéfices de l'architecture

L'attaque type décrite précédement :

- ne testera que les services de la zone semi-ouverte,
- ne pourra attaquer :
  - les démons sendmail... des Unix internes,
  - les services ou les chevaux de Troie NT.

Administration des stations

- Internes (très nombreuses) : laxisme tolérable
- Zone semi-ouverte (une poignée) : avec beaucoup de soins
  - versions à jour, correctifs, gestion utilisateurs,...

Évolutions possibles sans remise en cause de l'architecture :

- Nouveaux services réseau,
- Garde-barrière applicatif...

Alternatives à cette architecture

NAT

- Pourquoi pas ? Avantages :
  - oblige à faire un inventaire des services utilisés,
  - rend les stations clientes inaccessibles.
- Mais si bon inventaire et bon filtrage en place :
  - NAT n'apporte pas de fonction de sécurité supplémentaire,
  - incompatible avec certains protocoles (UDP).

Garde-barrière applicatif

- Goulot d'étranglement débit.
- Coût : il faut l'acheter et l'administrer.
- Mais solution envisageable sur certains sites.

Mise en place de l'architecture

- Ceci est un modèle à adapter
- Faut-il posséder « son » routeur ?
- On peut ouvrir si besoin particulier
- Où ? Porte du campus - du labo ?
- Définir une méthodologie pour la définitiion et la mise en place
  Concertation avec les utilisateurs

Ce modèle ne restreint pas l'utilisation de l'Internet
Un utilisateur a les mêmes services qu'avant

Bilan

- Réduction du nombre de machines/services visibles de l'internet
  $\rightarrow$ réduction du nombre de vulnérabilités.

- Connaître et maîtriser les flux dans le réseau.

- Évolution possible par ajout de fonctions.

- Il faut ensuite un suivi : journaux, ...

Exemple : au LAAS