Sécurité et **A**dministration des **S**ystèmes **I**nformatiques

Administration réseau

Fabrice Legond-Aubry
Fabrice.Legond-Aubry@u-paris10.fr

Les ressources WEB

- Adresses web:
  - www.ietf.org (Request For Comments - RFC)
  - www.iana.org, www.ripe.org (IP)
  - deptinfo.cnam.fr (cours réseaux)
  - www.linux-france.org/article/index.html (intro)
  - www.linux-france.org/prj/inetdoc (doc architecture réseau)
  - www.developpez.com (programmation & réseau)
- Livres:
  - R. Stevens, « Unix network programming », Prentice Hall, 1990
  - J-M. Rifflet et J-B. Yunès, « Unix : programmation et communication », Dunod
  - A. S. Tannenbaum, « Computer Networks », Prentice Hall.
  - W.R. Stevens, « TCIP/IP Illustrated, The protocols » , Addison Wesley
  - L. Toutain, « Réseaux locaux et Internet », Hermès

Vocabulaire

Introduction

- Attaque Réseau TCP/IP
  - Spoofing : Forger un message réseau faux et/ou malformé
  - Flooding : Inondation en vu de saturer une machine
  - Smurfing : Equivalent du flooding mais sur tout un réseau
  - Hijacking : Détournement d'un connexion
  - Sniffing : Ecoute des communications en vu d'obtenir des informations
  - Replay : Le rejeu
  - Denial Of Service : Déni de service

Les problèmes de sécurité dans la pratique

Introduction

- Internet n'a pas été conçu avec un objectif de sécurité
  - Internet est basé sur la confiance
  - Chaque niveau traversé par vos données offre des moyens d'attaques
- Internet est né avec les unix
  - il n'y a pas un UNIX mais une multitude d'implémentations différentes qui présentent toutes des caractéristiques propres
- Il existe de nombreux problèmes de sécurité dans la plus part des systèmes informatiques actuels.
- Au niveau physique et liaison (ethernet)
  - Sniffers qui écoute le réseau

Les problèmes de sécurité dans la pratique

Introduction

- Au niveau réseau (IP)
  - IP Spoofing et Smurfing
- Au niveau transport (TCP)
  - SYN Flooding
  - Au niveau applicatif (service réseau)
  - Déni de services (Deny of Services)
  - Buffer Overflows
- Attaque au niveau des services
- Attaque au niveau des personnes

Section : « Administration réseau »

Les bases du réseau

Les bases du réseau

Couche liaison et Routage IP

Attaques sur les couches basses

Couche Transport : TCP/UDP

Configuration réseau

Outils réseau

DHCP

DNS

Parefeu – NAT - SSL/TLS

IDS et Analyse

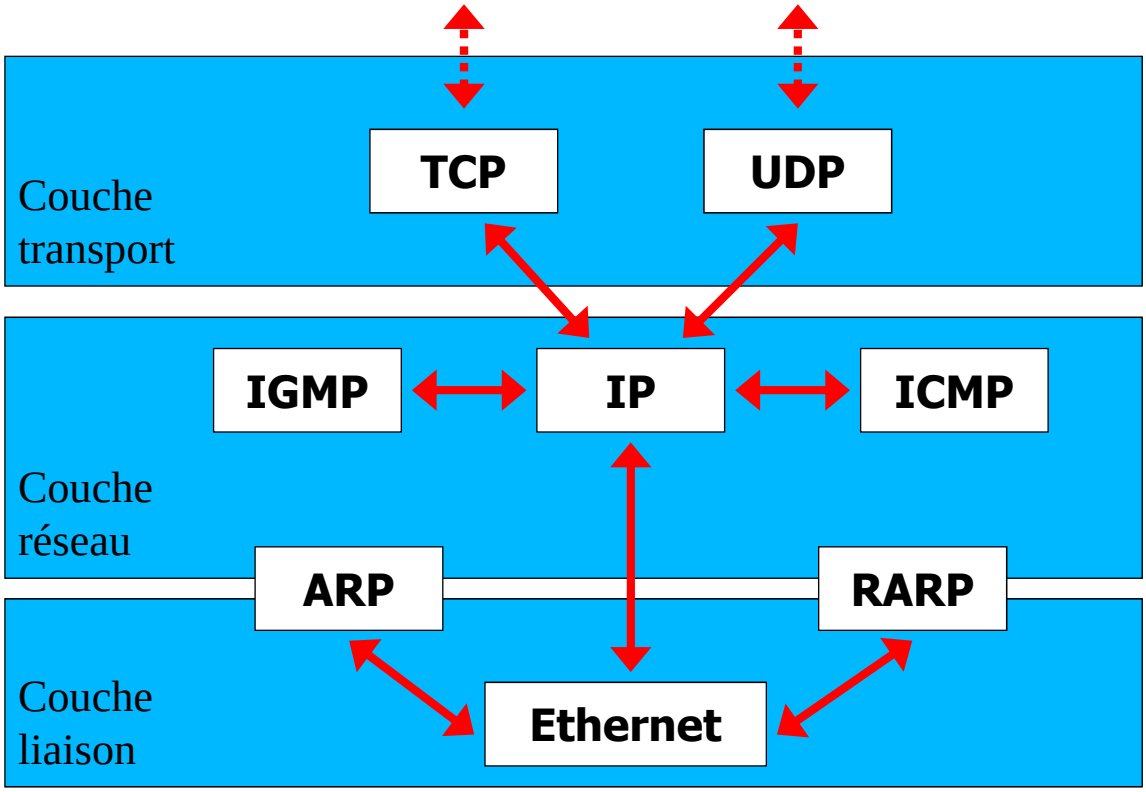
De la nécessité du réseau !

- 90% des services linux sont basés sur le réseau
- Vous vivez dans un monde interconnecté !!!
- **Revoir vos cours de réseau !!!!**
- Toute machine « Linux » a au moins un réseau: le réseau virtuel local « loopback »
- Une machine peut être connectée à plusieurs réseaux
- Il faut connaître les notions de réseaux pour
  - Définir l'architecture d'un parc de machines
  - Les interactions entres machines
  - Déployer, configurer, SECURISER un service

Pile TCP/IP

Couche 5-7 : application	Services linux
Couche 4 : transport (gestion des connexions)	TCP/UDP (désignation d'un processus)
Couche 3 : réseau (routage)	IP (désignation d'une machine)
Couches 1-2 : physique, liaison (transfert entre 2 machines reliés par une voie physique)	Ethernet , ATM, …

Relations entre les différents protocoles

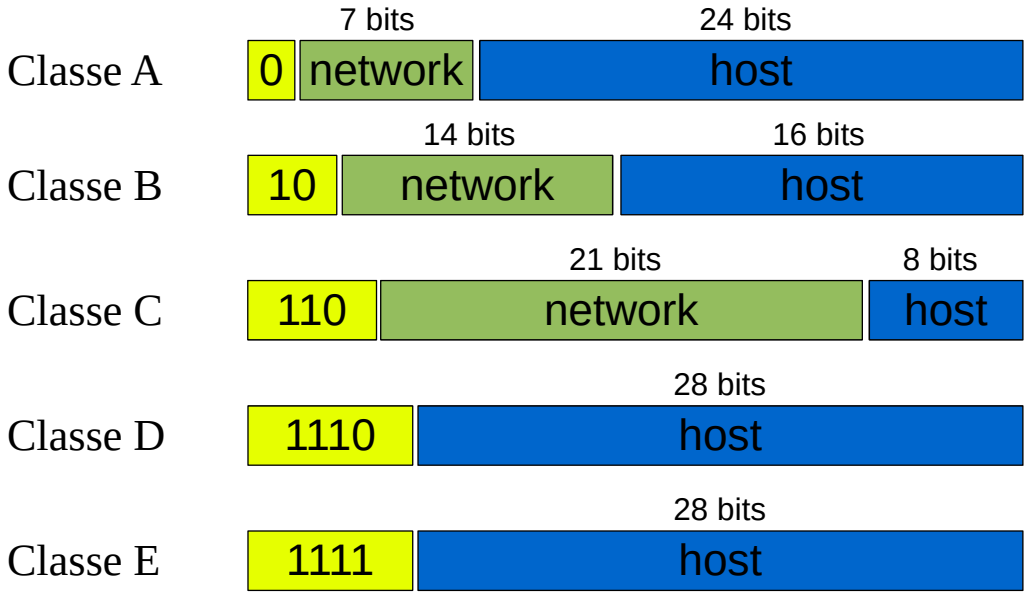


Gestionnaires des adresses IP

- Attribution par des organismes spéciaux :
  - IANA (www.iana.org et www.icann.org) centralise les affectations
  - RIPE (www.ripe.net) s'occupe des adresses européennes
  - AfriNIC (www.afrinic.net) s'occupe des adresses africaines
  - APNIC (www.apnic.net) s'occupe des adresses asiatiques et pacifiques
  - ARIN (www.arin.net) s'occupe des adresses de l'Amérique du nord
  - LACNIC (lacnic.net/en/index.html) s'occupe des adresses de l'Amérique latine et des Caraïbes

Notion de classe d'adresses IPv4

- Permet le routage et l'acheminement des données sur l'ensemble de l'internet
- IP permet de désigner une machine
- Notion de classes d'adresses (besoin de connaître le binaire !!!)



Nombres de réseaux par classe

- Classe A : 126 (2^7-2) réseaux possibles de 16 777 214 (2^{24}-2) machines
- Classe B : 16 382 (2^{14}-2) réseaux possibles de 65 534 (2^{16}-2) machines
- Classe C : 2 097 150 (2^{21}-2) réseaux possibles de 254 (2^8-2) machines
- Classe D : adresses de diffusion (multicast)
- Classe E : adresses réservées pour des usages futurs

Réseaux non routables

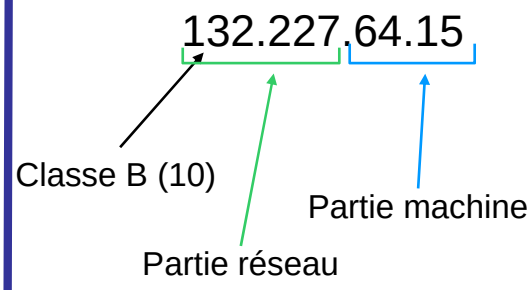
- Ce sont des réseaux qui ne seront jamais attribués à une entité
- Ils ne sont pas routable sur internet
- Ils sont réservés à un usage privé / interne :
  - 1 réseau de classe A : 10.0.0.0
  - 15 réseaux de classe B : 172.16.0.0 - 172.31.0.0
  - 255 réseaux de classe C : 192.168.0.0 - 192.168.255.0
- Aucun datagramme IP venant de l'extérieur ne doit porter ces adresses.

Les informations réseaux

- Une machine ayant l'IP : 193.22.143.52
  - ✔Adresse de classe C (193 commence par 110 en binaire)
  - ✔24 bits pour le réseau (network @)
  - ✔8 bits pour la machine (host @)
- Adresses particulières pour les réseaux de classes A,B,C
  - L'adresse « réseau » : Tous les bits d'adresse host à 0
    - ✔Exemple: 192.22.143.0
  - L'adresse de diffusion (broadcast) à tout le sous-réseau : Tous les bits d'adresse host à 1
    - ✔Exemple: 193.22.143.255
  - Le masque de sous réseau: Tous les bits d'adresse host à 0, tous les bits d'adresse réseau à 1
    - ✔Exemple: 255.255.255.0

Masque de réseau

- Le masque de réseau
  - Permet de séparer la partie réseau de la partie machine.
  - Possibilité de créer des sous-réseaux
- Exemples



Réseau LIP6, notations :

Masque: 255.255.0.0

@ réseau: 132.227.0.0

@ diffusion: 132.227.255.255

Sous-réseau SRC, notations :

Masque: 255.255.255.0

@ réseau: 132.227.64.0

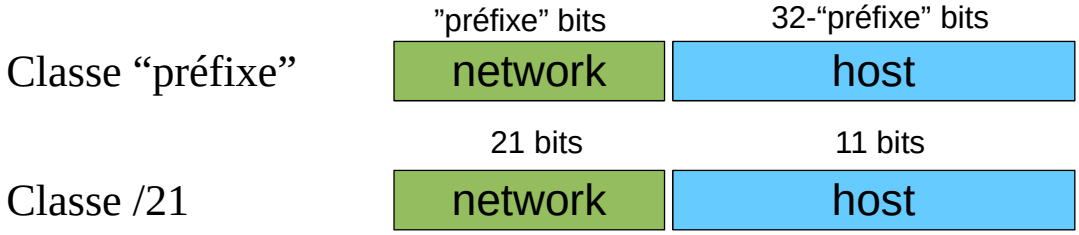
@ diffusion: 132.227.64.255

Saturation de l'espace d'adressage IPv4

- Pourquoi?
  - ✔Trop d'adresses distribuées par rapport au besoins (inutilisation)
  - ✔Pas de redistribution de la classe E, et des classes A?
  - ✔Sans doute 50% des adresses distribuées ne servent pas!
  - ✔Agrégation des classes C ➔ gonflement des tables de routages
- IPv6 : un espace d 'adressage beaucoup plus grand
  - ✔128 bits soit 16 octets au lieu de 32 bits soit 4 octets
  - ✔A priori 3,9 * 10^{18} adresses par mètre carré de surface terrestre
  - ✔Si l'on utilise très mal les adresses disponibles (comme dans le téléphone) ➔ 1500 adresses par mètre carré
- Autres solutions ?
  - ✔Les réseaux brûlés avec translation d'adresse (NAT) ?
  - ✔CIDR (Classless Inter-Domain Routing)

CIDR une solution en attendant IPv6

- Abandon de la notion de classe
- On définit les réseaux suivant les besoins
- Notation CIDR: adresse/préfixe (**RFC: 1517, 1518, 1519, 1520**)
- Pour construire un réseau de 2000 machines
  - Il faut 8 réseaux de classe C (/24) de 254 machines soit 2036 machines
  - Il faut 1 réseau de classe B (/16) de 65534 machines
  - Il faut 1 réseau CIDR /21 qui permet de déclarer 2046 machines (2^11-2)
- On agrège ainsi les réseaux pour une même entreprise
  - Par exemple, on peut agréger 2 réseaux de classes C (/24) en un réseau /23
- A la place de 3 classes, on utilise un préfixe :



Section : « Administration réseau »

Les bases du réseau

Routage IP et couche liaison

Attaques sur les couches basses

Couche Transport : TCP/UDP

Configuration réseau

Outils réseau

DHCP

DNS

Parefeu – NAT - SSL/TLS

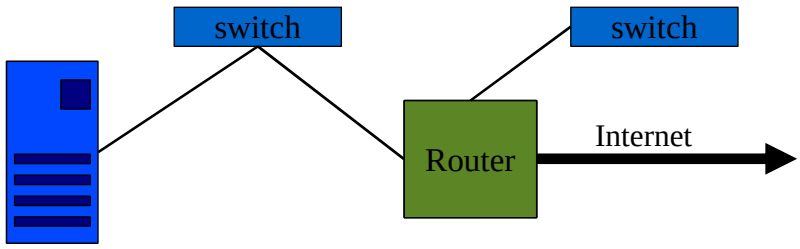
IDS et Analyse

Algorithmes de routage IP

- Routage sur un routeur (algorithme)
  - Recherche d'une destination correspondant à celle visée.
  - Recherche d'une entrée réseau où se trouverait le site visé (le plus proche).
  - Recherche d'une entrée de type défaut.
- Algorithmes de routage IP: OSPF, RIPv2, …
- Routage à partir d'une machine
  - Si le site à atteindre est connecté directement au site courant (par une liaison point à point ou en réseau local) ➔ le message est envoyé directement.
  - Sinon l'hôte dispose d'un routeur par défaut à qui il envoie tous les datagrammes qu'il ne peut acheminer.

Exemple: routage IP sur une machine

- Gateway: la route par défaut (default route)
  - Définit où envoyer tous les paquets qui ne sont pas destinés au réseau local




```
legond@hebe > netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
132.227.64.0    *               255.255.255.0   U         0 0          0 eth0
127.0.0.0       *               255.0.0.0       U         0 0          0 lo
default         castor          0.0.0.0         UG        0 0          0 eth0
```

- Ligne 1 : L'accès au réseau local (ethernet) de l'hôte
- Ligne 2 : La boucle locale (loopback) pour les messages qui ne sortent pas du site
- Ligne 3 : L'accès à un routeur par défaut qui permet de passer sur l'internet

Un exemple de configuration réseau

```
legond@morphee > ifconfig -a
eth0      Link encap:Ethernet  HWaddr 00:30:13:3D:2B:65
          inet addr:132.227.64.42  Bcast:132.227.64.255  Mask:255.255.255.0
          inet6 addr: fe80::230:13ff:fe3d:2b65/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3953229 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2616429 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2991794751 (2853.1 Mb)  TX bytes:2551152611 (2432.9 Mb)
          Base address:0x2000 Memory:e8100000-e8120000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:56199 errors:0 dropped:0 overruns:0 frame:0
          TX packets:56199 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:118600424 (113.1 Mb)  TX bytes:118600424 (113.1 Mb)
```

Type couche liaison

Adresse machine

Adresse broadcast

Adresse Couche liaison (@MAC)

Masque réseau

Lien couche liaison (ethernet) / réseau (IP)

- Liaison entre la couche réseau (@IP) et la couche liaison ethernet (@MAC)
  - Utilisation des protocoles ARP et RARP
  - Dès qu'une machine a besoin de savoir à quelle @MAC correspond une @IP, elle diffuse une demande de correspondance sur le réseau physique

```
root@scylla >  tcpdump -f -i eth0 arp or rarp
01:52:55.652713 arp who-has eros tell scylla
01:52:55.652910 arp reply eros is-at 00:c0:4f:89:d0:6c
```

  - Le résultat est mise en cache, jusqu'à la détection d'une erreur

Lien couche liaison (ethernet) / réseau (IP)

- Gestion du cache des couples @IP/@MAC
  - « arp » affiche et manipule les informations de la table

```
root@scylla > arp -a
morphee (132.227.64.42) at 00:30:13:3D:2B:65 [ether] on eth0
castor (132.227.64.15) at 00:10:0D:3D:C4:00 [ether] on eth0
```

- Fixer des couples @IP/@MAC
  - Pour éviter des attaques
  - « arp –s @ip @mac »
  - « arp –f nom_de_fichier » pour une liste de couples

Lien couche liaison (ethernet) / réseau (IP)

- Gestion du cache des couples @IP/@MAC
  - « arping » permet d’envoyer des requêtes arp/rarp

```
root@scylla > arping diane
ARPING 132.227.64.48 from 132.227.64.30 eth0
Unicast reply from 132.227.64.48 [00:11:95:22:03:30]  0.674ms
Unicast reply from 132.227.64.48 [00:11:95:22:03:30]  0.653ms
```

  - « arping » permet de détecter deux machines ayant la même ip

```
root@scylla > arping -D ares
ARPING 132.227.64.31 from 0.0.0.0 eth0
Unicast reply from 132.227.64.31 [00:0F:B5:47:10:59] for
     132.227.64.31 [00:0F:B5:47:10:59] 0.639ms
Sent 1 probes (1 broadcast(s))
Received 1 response(s)
```

Section : « Administration réseau »

La base du réseau

Couche liaison et Routage IP

Attaques sur les couches basses

Couche Transport : TCP/UDP

Configuration réseau

Outils réseau

DHCP

DNS

Parefeu – NAT - SSL/TLS

IDS et Analyse

Type de paquets principaux circulant sur IP

- Paquets UDP
  - émissions en mode **non** connecté
  - Permet la transmission point-à-point et la diffusion (broadcast)
- Paquets TCP
  - Émissions en mode connecté
  - Transmissions point-à-point exclusivement avec qualité de transmission
- Paquets de gestion du réseau ICMP/IGMP
- Adresse de diffusion :
  - Utilisé pour envoyer des paquets sur tout le réseau local

Il faut savoir configurer vos routeurs pour ne pas diffuser tous les paquets de broadcast!

Utilité de la couche transport

- TCP et UDP permettent la discussion entre des services (des processus)
- TCP/UDP permet de désigner un processus sur une machine
- On désigne un processus par un numéro de port
- Il existe des ports officiels associés à un type particulier de service
- Il existe des ports libres qui peuvent être associés à des applications « utilisateur »
- La liste officiel des correspondances ports/service
  - Fichier « ***/etc/services*** »
  - Sur le web : http://www.iana.org/assignments/port-numbers
  - Sur les sites de sécurité pour les ports « suspects » (iss.net, neophasis)

Les ports

- Les ports 1 à 1023 sont privilégiés : ils ne peuvent être ouverts qu'avec les droits « root ».
- Les ports 1024 à 65535 sont non privilégiés (ou éphémères) et peuvent être ouvert par tous les utilisateurs.
- Un processus peut ouvrir plusieurs ports
- Un ports ne peut être contrôlé que par un processus
- Contenu du fichier « ***/etc/services*** » :

```
...
ftp-data        20/tcp
ftp-data        20/udp
ftp             21/tcp          # File Transfert Protocol
ssh             22/tcp          # SSH Remote Login Protocol
telnet          23/tcp
...
```

- Pour la sécurité et le contrôle d'accès
  - Contrôle d'accès aux services réseaux par « ***/etc/hosts.allow*** » et « ***/etc/hosts.deny*** »

Les bases du réseau

Routage IP et couche liaison

Couche Transport : TCP/UDP

Configuration réseau

Outils réseau

Parefeu – NAT - SSL/TLS

DHCP

DNS

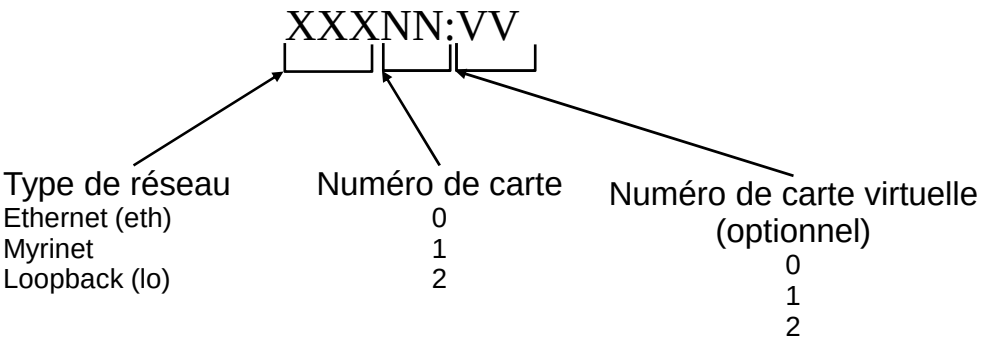
IDS et Analyse

Une machine, plusieurs prises réseaux

- Une machine peut avoir plusieurs cartes réseaux
- Un maximum de 64 cartes par machines
- Pourquoi faire ?
  - Routeurs, Firewalls
  - Transferts entre réseaux
  - Agrégation de liens, répartition de charge, tolérance aux fautes
- On peut donner plusieurs IP à une même carte réseau
  - IP Virtuelles
  - Exemple: Serveur Web avec HTTPS
  - Exemple: Serveur Web avec des sites virtuels

Le nom des cartes réseaux

- Linux supporte jusqu'à 256 adresses virtuelles



Exemples (Ethernet):

eth0	device = /dev/eth0
eth1	device = /dev/eth1
eth0:1	device = /dev/eth0

Fichiers de configuration réseaux

- **Fichiers de correspondance sur la machine**
  - ***/etc/hosts*** **(correspondance IP/nom d'hôte au boot seulement)**
    - ✔**127.0.0.1 localhost**
    - ✔**137.194.160.21 horla**
  - ***/etc/rpc*** **(correspondance nom de procédure/n° de procédure)**
  - ***/etc/networks*** **(correspondance nom de réseau/n° de réseau)**
  - ***/etc/protocols*** **(correspondance nom de protocole/n° de protocole)**
  - ***/etc/ethers*** **(correspondance IP/n° Ethernet)**
  - **Le fichier «** ***/etc/resolv.conf*** **» permet de définir comment lier une IP et un nom de machine (par fichier ou par DNS)**
- Configuration dépend du système :
  - Mandrake : fichier « ***/etc/sysconfig/network-scripts/ifcfg-eth0*** »
  - Gentoo : fichier « ***/etc/conf.d/net*** »
  - …

Contenu des fichiers configurations

- Mandrake « /etc/sysconfig/network-scripts/ifcfg-eth0 »

```
DEVICE=eth0
BOOTPROTO=static
IPADDR=132.227.64.30
NETMASK=255.255.255.0
NETWORK=132.227.64.0
BROADCAST=132.227.64.255
ONBOOT=yes
```

- Gentoo « /etc/conf.d/net »

```
iface_eth0="132.227.64.31 broadcast 132.227.64.255
netmask 255.255.255.0"
gateway="eth0/132.227.64.15"
```


Outils de configuration réseaux

- **Scripts de démarrage / arrêt** du réseau :
  - Sur Mandrake : fichier « ***/etc/init.d/network*** »
  - Sur Gentoo : fichier « ***/etc/init.d/net.eth0*** »
  - Il configure automatiquement les routes
- La configuration de la carte se fait par « ***ifconfig*** » ou « ***ethtool*** »
- L'affichage et la manipulation de la table de routage IP se fait par la commande « ***route*** »
- L'ensemble du contrôle TCP/IP peut se faire par la commande « ***ip*** »

Section : « Administration réseau »

Les bases du réseau

Couche liaison et Routage IP

Couche Transport : TCP/UDP

Attaques sur les couches basses

Configuration réseau

Outils réseau

DHCP

DNS

Parefeu – NAT - SSL/TLS

IDS et Analyse

Niveau Physique : Ecoute des supports physiques

- Ethernet physique
  - Plug sur le câble
    - ✔Introduction de bruit
    - ✔Ecoute du support physique
  - Emissions électromagnétiques des câbles
- Moyens de lutte
  - Blindage des câbles, cage de faraday
  - Filtrage électrique
  - Ne pas connecter les machines vers l'extérieur
  - Contrôle des câbles par du matériel spécifique

Attaques sur les couches basses

Niveau Physique : Réseau sans fils 802.11

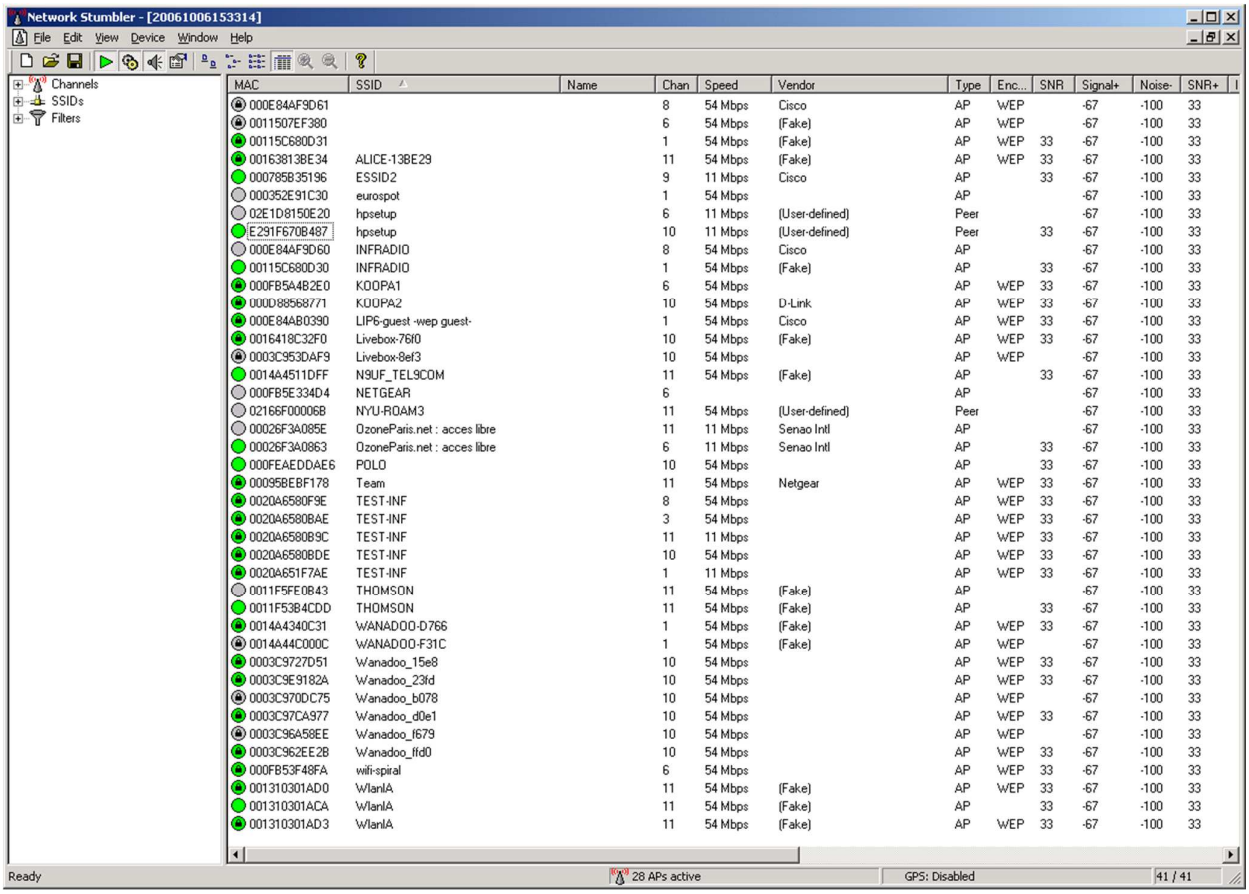
- Le 802.11 aka WiFi (Wa fa)
  - C'est un standard de réseau sans fil
  - Il existe une pléthore de protocoles
  - On parle de l'alphabet 802.11 (802.11a, 802.11b, …)
  - Les seuls a retenir sont 802.11g, 802.11i, 802.11x
- Le 802.11 soulève des problèmes
  - Qui existent déjà !!!! ➔ Mise en exergue des problèmes filaires !
  - Périmètre de sécurité
    - ✔Portables d'inconnus entre dans le périmètre
    - ✔Equivalent à une prise de la taille d'une sphère de 80m de rayon
- Sans cryptage (confidentialité) et sans authentification
  ➔ SUICIDAIRE !!!
- Ecoute et brouillage possible

Attaques sur les couches basses

Niveau Physique : 802.11 Moyens de lutte

- Contrôle d'accès (efficacité limitée)
  - Spatial: mesures et calibrage des puissances du signal des bornes
  - Par adresse: Contrôle d'accès des adresses MAC
  - Pas de diffusion du SSID !
- Confiner ce réseau dans un réseau spécial externe
  - Eviter les accès IP sur le réseau interne
- Confidentialité: Le WEP une coquille vide ➔ A JETER
  - Faiblesse du chiffrement, pas de gestion des clefs
  - airsnort.shmoo.com, www.cr0.net:8040/code/network/aircrack (dans beaucoup de distrib linux)
- Confidentialité et authentification ➔ la seule solution valable, ENCORE EN DEPLOIMENT !
  - **802.11i WPA2 (et PAS WPA simple ou de WEP) ➔ confidentialité**
  - **802.11x avec un serveur radius (www.freeradius.org) et des modules EAP**
- Audit
  - Journalisation des adresses inconnues (MAC et IP)
  - Journalisation des scan
  - Détection des réseaux pirates internes et externes
  - Recherche de signal en bordure (istumbler.net et autres) et triangularisation
- Saturation hertzienne de la zone couverte

Niveau Physique : Les AP WiFi autour de mon bureau



Ethernet : Rappels écoute réseau (sniffing)

- But:
  - Collecte d'informations sur les données circulant sur le réseau
  - Analyse à posteriori des trames
  - Attaque de décryptage sur les données (analyse différentielle)
  - Captures des mots de passe en clair (POP3, TELNET, IMAP, …)
- Fonctionne avec tous les protocoles de niveau supérieur
  - La capture s'effectue au niveau 2 avec du matériel classique
  - La capture s'effectue au niveau 1 avec du matériel spécialisé

Attaques sur les couches basses

Ecoute ethernet: fonctionnement

- Ethernet c'est 90% des réseaux locaux
  - Prix ridicules, déploiement aisé, grande variété de matériel
  - Ethernet est un support DIFFUSANT !!!! ➔ facilité d'écoute !!!!!
- Il faut passer la carte en mode "Promiscuous"
  - Permet à la carte de capturer tout ou partie des paquets qui transitent sur le réseau local
    - ✔ Même si les paquets sont non destiné à l'adresse IP de la machine qui écoute
    - ✔ Ne permet pas la capture hors du réseau local
  - Peut être filtrer par les routeurs
- Accès simple
  - Librairie PCAP
  - Raw socket: packet_socket = socket(PF_PACKET, int socket_type, int protocol);
- Utilisation de logiciels d'écoute
  - Très facile à utiliser
  - TCPDUMP (Linux/Windows), ethereal, wireshark (Windows/Linux)
  - Logiciels spécialisés (dnsiff), Distributions linux spécialisées (BackTrack)
  - Network Associates Sniffer (Windows)

Attaques sur les couches basses

Ethernet : Détection locale des écoutes

- Détection possible des cartes en mode "promiscous"
- Vérification locale
  - Un rootkit peut cacher des informations
  - Un rootkit peut en cacher un autre
  - Ne pas utiliser les programmes de la machine !
  - Télécharger ses propres programmes compilés !!!
  - Ifconfig:

```
pollux 14:46 >ifconfig eth0
  eth0  Link encap:Ethernet  HWaddr 00:C0:4F:24:27:E7
        inet addr:132.227.64.49  Bcast:132.227.64.255
  Mask:255.255.255.0
  UP BROADCAST RUNNING PROMISC MULTICAST  MTU:1500  Metric:1
   RX packets:9566866 errors:44 dropped:0 overruns:0 frame:4
   TX packets:7763589 errors:0 dropped:0 overruns:0 carrier:0
   collisions:0 txqueuelen:100
  Interrupt:14 Base address:0xcc00
```

  - Un processus root inconnu est en cours d'exécution (ps)
  - Vérifier les comportements des programmes avec lsof et strace

Détection distante des écoutes (antisniff)

- En théorie c'est impossible
  - les nœuds sont passifs ➔ ils ne transmettent rien
- Dans la pratique, c'est parfois possible
  - Difficile à détecter !!
- Méthode de ping 1
  - Si la machine qui a pour @IP 132.227.64.234 et pour @MAC aa:bb:cc:dd:ee:ff est suspectée d'écouter le réseau.
  - On émet une demande ICMP "echo request" en modifiant l'adresse MAC (ie: aa:bb:cc:dd:ee:f0)
  - Si la machine répond, elle était en mode d'écoute
    - ✔Le mode « promiscuous désactive » le filtre de l'@ MAC et répondra à l'@IP sans vérifier l'adresse MAC

Antisniff: autres méthodes

- Méthode du DNS
  - Les machines qui écoutent peuvent faire des requêtes DNS
  - Faire un ping et vérifier les demande DNS arrivant de machine non connues (@IP mauvaise) !!
- Méthode de routage-source
  - Envoyer un paquet à une machine intermédiaire en demandant son acheminement à la machine suspectée
  - Si la machine intermédiaire ne fait pas suivre le paquet et si la machine suspectée répond, elle écoute le réseau

Ethernet: Antisniffing - autres méthodes

- Méthode de leurre
  - On génère du traffic POP, TELNET, FTP … avec des comptes fictifs (sans réel droits)
  - On vérifie si des login sont effectués sur ces comptes
- Méthode de la latence
  - On génère un traffic ethernet important
    - ✔Il sera filtré par les machines normales
    - ✔Il sera capturé par les machines en écoute
  - On ping les machines et on mesure leur temps de réponse
- Outils: antisniff, CPM (check promiscusous method), neped

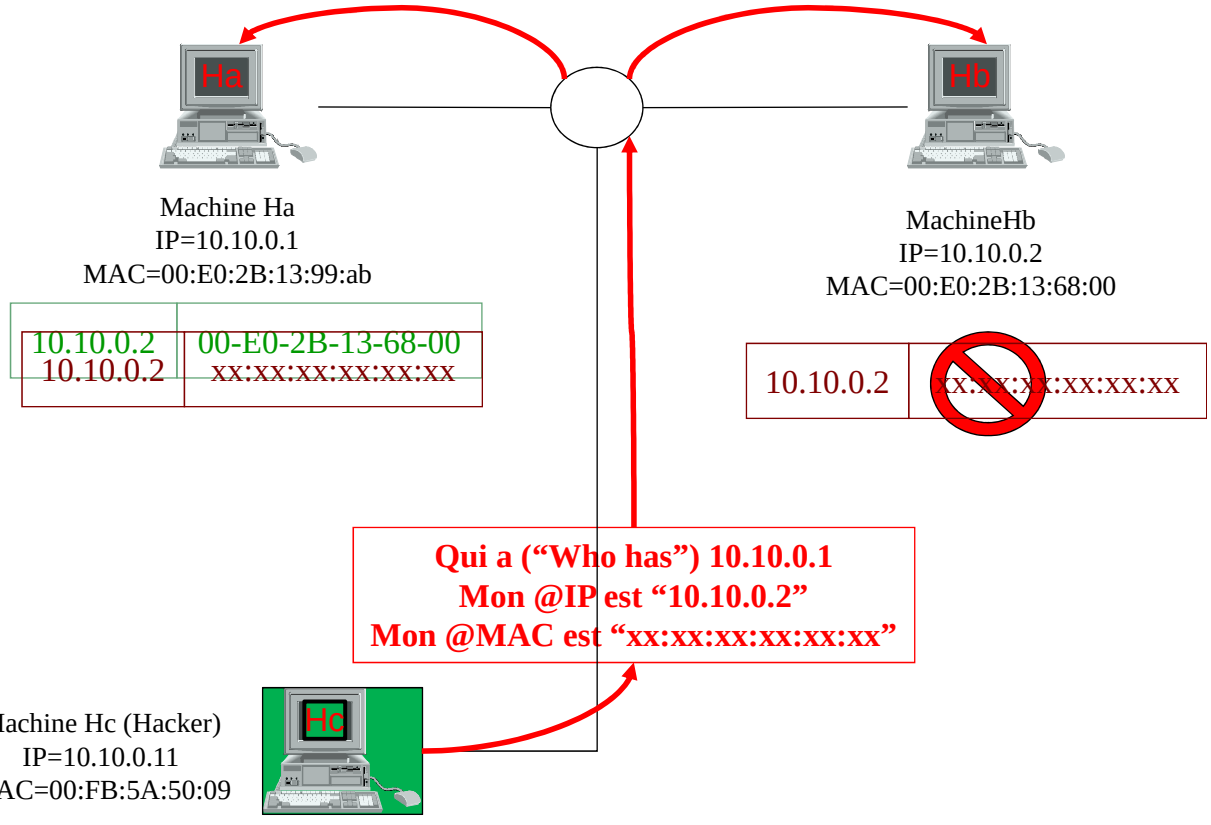
Lutter contre les écoutes

- Eviter la capture de mot de passe
  - Eviter l'authentification et la circulation des données en clair
    - ✔Utiliser le cryptage sur les couches basses (SSL,IPSEC)
    - ✔Utiliser l'encapsulation applicative (SSH, stunnel)
- Recherche systématique des machines inconnues
  - Découverte du réseau (HP openview, netdisco, …)
- Limiter la connectivité des machines aux machines connues
  - ACL avec adresses ethernet au niveau des switchs
  - VMPS sur les switch (correspondance @MAC/vlan)
  - Puces TPM por l'authentification fortes des machines

Ethernet : Attaques ARP

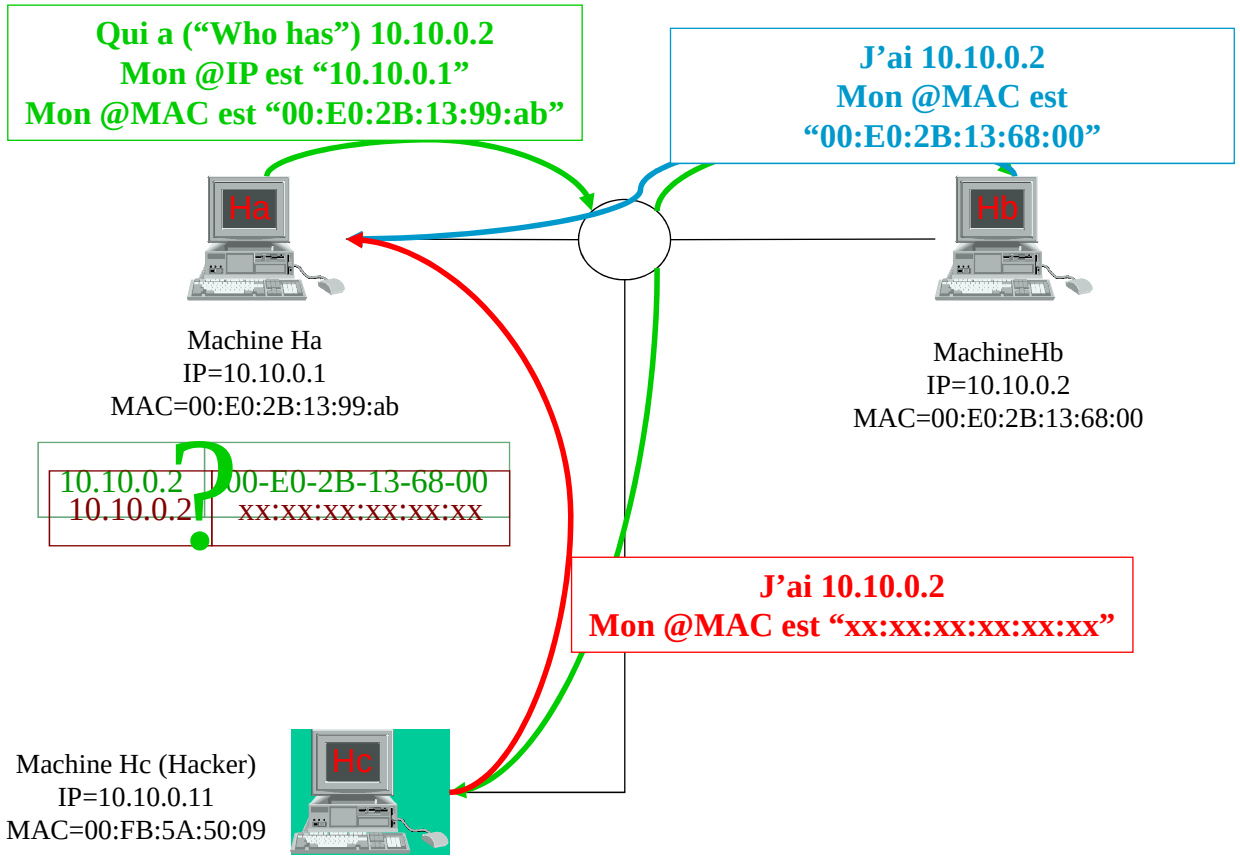
- ARP-RARP ➔ lien entre @MAC et @ IP
- ARP maintient un cache des machines qui diffusent sur le réseau
- But des attaques ARP:
  - Détourner le traffic réseau vers sa machine
  - En particulier, remplacer le couple (@MAC/@IP) du routeur par sa propre machine
  - Déni de service
- Moyen:
  - Poisoning : créer de fausses entrées dans les caches ARP
  - Flooding: Saturer les tables ARP
  - Cloning: imiter l'adresse MAC d'une autre machine
  - Spoofing: Forger de fausses réponses ARP
- Utilitaires:
  - http://web.syr.edu/~sabuer/arpoison/
  - http://ettercap.sourceforge.net/ (logiciel d'attaque ARP, SSH, tueur de connexion)
  - http://www.thehackerschoice.com/releases.php (génère des fausses réponses ARP)

Etherenet : Attaque ARP - Poisoning par diffusion



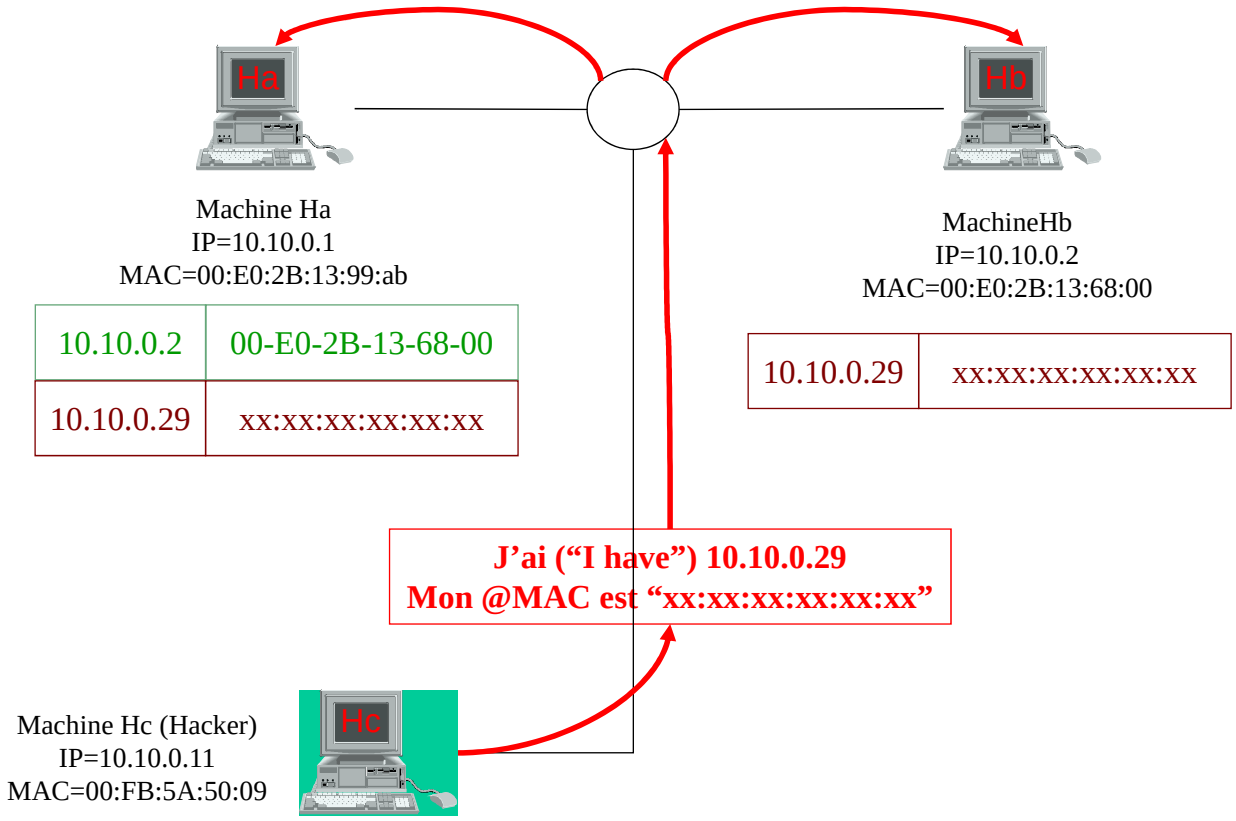
Attaques sur les couches basses

Ethernet : Attaque ARP - Poisoning par requête

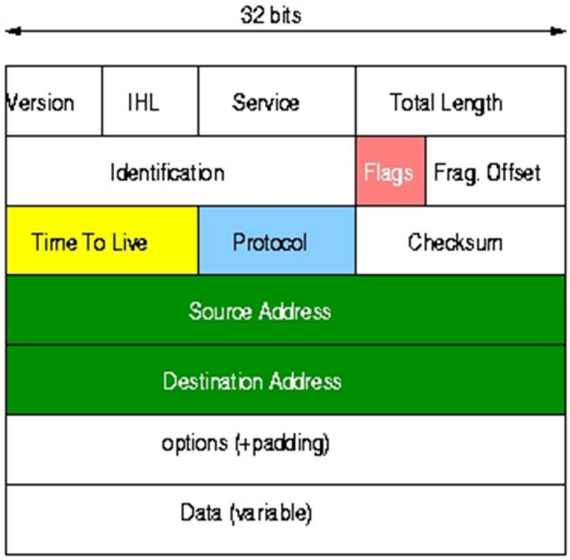


Attaques sur les couches basses

Ethernet : Attaque ARP - Poisoning par diffusion de réponse



Le protocole IP



- IP suppose que
  - Les @ IP source et destination sont fiables
  - Le TTL est utilisé pour éviter que des paquets circulent indéfiniment dans le réseau (ghost packet)
  - Protocoles: ICMP=1, IGMP=2, TCP=6, UDP=17, RAW=255
  - Checksum ➔ contrôle CRC
  - Permet des options qui peuvent poser problème
    - ✔Source Routing
    - ✔Bit DF/MF
  - Données: taille max. 64ko

Attaque IP : Spoofing

- En générale, il s'agit d'une attaque aveugle !
- Rappel: Internet est basé sur la confiance.
- Modification du champ de l'@IP source et de certaines options
- But:
  - Usurpation d'identité ➔ on se fait passer pour une autre machine
  - Passer à travers les filtres IPs de certaines machines
- La (véritable) source:
  - peut recevoir les réponses si elle est local (sniffing)
  - NE peut PAS recevoir de réponse si elle est distante
    - ✔Sauf cas exceptionnel (IP source routing)
- A Coupler avec des attaques TCP (cf. prochaine section)

Attaque IP : Spoofing

- Détection: Peu évidente
  - Analyser les logs du parefeu
    - ✔ Connexions inhabituelles, violation des ACL, paquets rejetés
- Lutte contre le spoofing IP:
  - Ne pas se limiter à des ACL basés sur les @IP
  - Implanter des règles strictes sur les routeurs
    - ✔ Tout paquet provenant de l'extérieure ne peut avoir une @IP source interne
    - ✔ Tout paquet provenant de l'extérieure ne peut avoir une @IP source non attribué ou non routable
      - Constitution d'une liste noire des @IP non attribuée à partir des liste de l'IANA
    - ✔ Tout paquet allant à l'extérieur ne peut avoir @IP source n'appartenant pas à votre réseau
  - Associer les @MAC avec les @IP pour les machines critiques
    - ✔ ie: LES SERVEURS (en particulier ceux d'authentifications) !!
    - ✔ PB: les entrées ARP statiques sont parfois mal supportées
  - Sonde de monitoring des couples (@MAC,@IP)
    - ✔ Détection des cas de divergences avec la conformité et des trames ARP anormales
    - ✔ Alerte + Engagement des contres mesures (détection, analyse, blocage)

Les protocoles ICMP/IGMP

- Le protocole ICMP (Internet Control Message Protocol)
  - Le protocole ICMP est utilisé par tous les routeurs
  - Il est utilisé pour signaler une erreur sur une machine connectée
- Message ICMP = Type (8 bits), Code (8 bits), Checksum (16 bits), Message (taille variable)
- Couples Type / Message :
  - 0: Echo reply, 8: Echo request
  - 3: Destination Unreachable,
  - 5: redirect, 6: alternate host address, ...

Moyen de lutte : IPSEC - VPN

- Permet d'éviter le spoofing @IP et le sniffing
- IP-Secure / Développé par l'IETF (RFC 2401, 2402, 2406, 2409, 2411)
  - Utilisé pour implanter les VPN (Virtual Private Network)
  - Solution niveau 3 (réseau) ➔ IP
  - Pas une solution de niveau "applicatif" comme SSH
- IPSEC fournit :
  - La communication est crypté de bout en bout
  - L'authentification forte, confidentialité et intégrité
  - Indépendant de TCP/UDP, repose sur IP
- Il est supporté :
  - en natif par beaucoup de système et par la grande majorité des routeurs
  - par des drivers/applications sur beaucoup d'autres systèmes
- Mode IP-Sec: 2 mode de transport
  - Payload ➔ seulement les données sont cryptées 'encapsulation des données)
  - Tunnel ➔ toute la communication est encryptée (encpasulation totale)
- Mode IP-Sec: 2 protocoles
  - AH ➔ qui ne permet que l'authentification forte et intégrité, pas de confidentialité
  - ESP ➔ qui permet la confidentialité

Section : « Administration réseau »

Les bases du réseau

Routage IP et couche liaison

Couche Transport : TCP/UDP

Configuration réseau

Outils réseau

Parefeu – NAT

DHCP

DNS

IDS et Analyse

Outils réseau

Diagnostiques réseaux : ping, netstat, telnet

- La commande « ***ping nom*** » permet de savoir si une machine est vivante. Une version parallèle nommée « *fping* » existe.
- La commande « ***netstat*** » permet de savoir l'ensemble des ports ouverts

```
root@scylla >  netstat -ln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:32768           0.0.0.0:*               LISTEN
tcp        0      0 127.0.0.1:994           0.0.0.0:*               LISTEN
......
```

- « ***netstat –r*** » affiche la table de routage
- « ***host nom*** » permet d'obtenir l'IP ou le nom de la machine
- « ***resolveip ip*** » permet d'obtenir l'ip d'une machine (ou d'une IP)
- « ***telnet machine port*** » permet d'ouvrir sur une connexion sur un service d'une machine distante
- « ***clockdiff*** » permet d'obtenir le décalage temporel entre deux machines
  - tickets Kerberos
  - *clockskew* lors de compilations sur NFS

Diagnostiques réseaux : lsof

- Une source d'information importante !!!!
- Obtenir la liste des fichiers ouverts par le processus 1200
  - « ***lsof -p 1200*** »
- Obtenir la liste des ports ouverts par le processus 1200
  - « ***lsof -p 1200 -i 4 –a*** »
- Savoir quel(s) processus sont en contact avec les ports 1 à 1024 de ares.lip6.fr
  - « ***lsof –i @ares.lip6.fr:1-1024*** »
- Savoir quel(s) processus ont ouvert le fichier « ~/foobar »
  - « ***lsof ~/foobar*** »
- Savoir quels sont les fichiers ouverts par l'utilisateur « apache »
  - « ***lsof –u apache*** »

Informations sur le réseau : dig/nslookup

- Interroger un dns pour obtenir un nom de machine ou une ip : « ***nslookup*** »
- « ***dig*** » est identique à nslookup mais il offre plus d'options

```
legond:@eos > nslookup www.lemonde.Fr
Server:         132.227.64.13
Address:        132.227.64.13#53

Non-authoritative answer:
www.lemonde.Fr  canonical name = www.lemonde.fr.d4p.net.
www.lemonde.fr.d4p.net  canonical name = a245.g.akamai.net.
Name:   a245.g.akamai.net
Address: 193.50.203.46
Name:   a245.g.akamai.net
Address: 193.50.203.53
```


Informations sur le réseau

- « ***traceroute*** », « ***traceroute6*** », « ***tracepath*** » et « ***tracepath6*** » permettent de voir le chemin jusqu'à une machine

```
legond@scylla > tracepath  www.lemonde.fr
 1:  scylla (132.227.64.30)                                0.258ms pmtu 1500
 1:  castor (132.227.64.15)                                1.519ms
 2:  r-jusren.reseau.jussieu.fr (134.157.254.126)          1.557ms
 3:  gw-rap.rap.prd.fr (195.221.127.181)                 asymm  4   2.292ms
 4:  jussieu-g0-1-165.cssi.renater.fr (193.51.181.102)     2.541ms
 5:  nri-c-pos2-0.cssi.renater.fr (193.51.180.158)         2.364ms
 6:  193.50.203.53 (193.50.203.53)                         3.787ms reached
     Resume: pmtu 1500 hops 6 back 6

legond@scylla > traceroute  www.lemonde.fr
traceroute to a245.g.akamai.net (193.50.203.53), 30 hops max, 38 byte packets
 1  castor (132.227.64.15)  1.334 ms  1.211 ms  1.387 ms
 2  r-jusren.reseau.jussieu.fr (134.157.254.126)  0.821 ms  1.305 ms  0.614 ms
 3  gw-rap.rap.prd.fr (195.221.127.181)  1.760 ms  1.672 ms  1.545 ms
 4  jussieu-g0-1-165.cssi.renater.fr (193.51.181.102) 1.343 ms 0.790 ms 1.184 ms
 5  nri-c-pos2-0.cssi.renater.fr (193.51.180.158)  1.617 ms  1.605 ms  1.479 ms
 6  193.50.203.53 (193.50.203.53)  1.911 ms  1.921 ms  0.893 ms
```

Analyser le réseau

- Commandes SunOS : ***etherfind***, ***snoop***
- « ***tcpdump*** » permet la capture et le filtre des communications entre les machines et/ou les services
  - Capturer toutes les communications provenant de zeus
    - ✔« ***tcpdump src host zeus*** »
  - Capturer tous les paquets provenant du serveur DHCP
    - ✔« ***tcpdump udp and port 67*** »
  - Capturer les paquets de gestion du réseau (ICMP)
    - ✔« ***tcpdump icmp*** »
- « ***ethereal*** » permet la capture et l'analyse du trafic réseau.
  - Des exemples ? RTFM !!
  - Très complexe, Très puissant

Plan de cours

La base du réseau

Routage IP et couche liaison

Couche Transport : TCP/UDP

Configuration réseau

Outils réseau

Parefeu – NAT

DHCP

DNS

IDS et Analyse

Politiques de filtrages

- Le filtrage est un des outils de base de la sécurité. IL EST NECESSAIRE !
- Filtrage optimiste : PERMIT ALL
  - Tout est permis à part quelques services (ports)
  - Facile à installer et à maintenir
    - ✔ Seulement quelques règles à gérer
  - Sécurité faible
    - ✔ Ne tient pas compte des nouveaux services pouvant être ouvert
    - ✔ Ex: un utilisateur ouvre un serveur ftp personnel, …
- Filtrage pessimiste : DENY ALL
  - Rejet systématique
    - ✔ Exception : services spécifiques sur machines spécifiques
    - ✔ Ex: Autorisations explicites pour les services HTTP, SMTP, POP3, …
  - Plus difficile à installer et à maintenir
    - ✔ En particulier pour certains services (ex: FTP)
  - Sécurité forte
    - ✔ Les nouveaux services doivent être déclarés
- Prendre en compte les connexions entrantes et les connexions sortantes

Filtrage sur les routeurs

- Installer des règles sur les routeurs pour empêcher certains trafic de passer par les routeurs
  - Utilisation des Access Lists (Cisco, ...)
  - Le filtrage peut être fait sur les critères suivants :
    - ✔Par protocoles ([ethernet], IP, ICMP, TCP, UDP, ...)
    - ✔Par adresses (suivant le protocole)
    - ✔Par les numéros de port TCP/UDP (HTTP, SMTP, POP3, ...)
    - ✔Par masque d'adresse
    - ✔Par les interfaces d'accès
    - ✔Structure et/ou contenu des paquets
  - Attention à l'ordre des règles
    - ✔La première qui correspond est celle sélectionnée (Fist Matching, First Applied!)
  - Une politique doit être installée

Filtrage sur les machines clientes

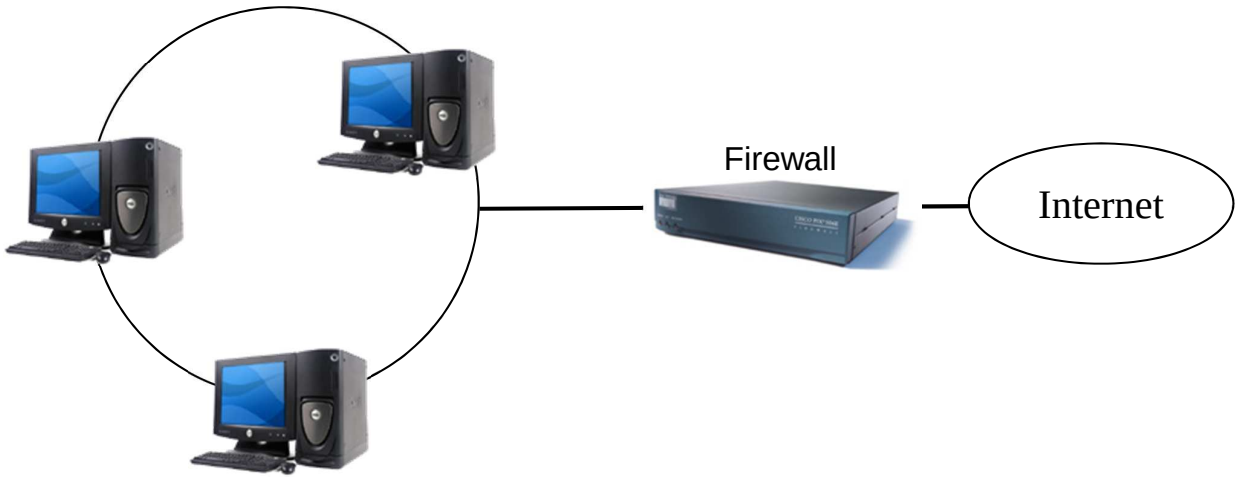
- Le filtrage doit aussi être fait aux niveaux des machines
- Les même type de politiques peuvent être appliquées
  - Optimistes ou Pessimistes
  - Les mêmes critères de filtrages peuvent être appliqués
  - De nouveau critères peuvent être ajoutés
    - ✔Contrôle des utilisateurs et des applications
- Problèmes :
  - Difficultés de maintenance (prévoir un déploiement automatique)
  - Une MAJ doit être déployée
- Doit être adaptables sur les portables (migration)

Firewalls / Routeur

- Différence entre un routeur et un firewall
  - Un firewall ne fait pas de “IP FORWARDING”
  - Un firewall peut faire du routage au niveau applicatif
    - ✔Existence de mandataires HTTP, POP3, etc …
- Les mandataires peuvent être intelligent
  - Filtrage par le contenu (informations)
  - La forme des paquets
- Implantation
  - Un matériel spécialisé (Cisco PIX, …)
  - Une machine simple avec plusieurs cartes réseaux + logiciels
    - Firewall 1 (Checkpoint), Raptor, Shorewall (Linux), ...

Architecture avec Firewall sans routeur

- Modèle avec double réseau
  - Pas de routage IP
  - Filtrage applicatif seulement

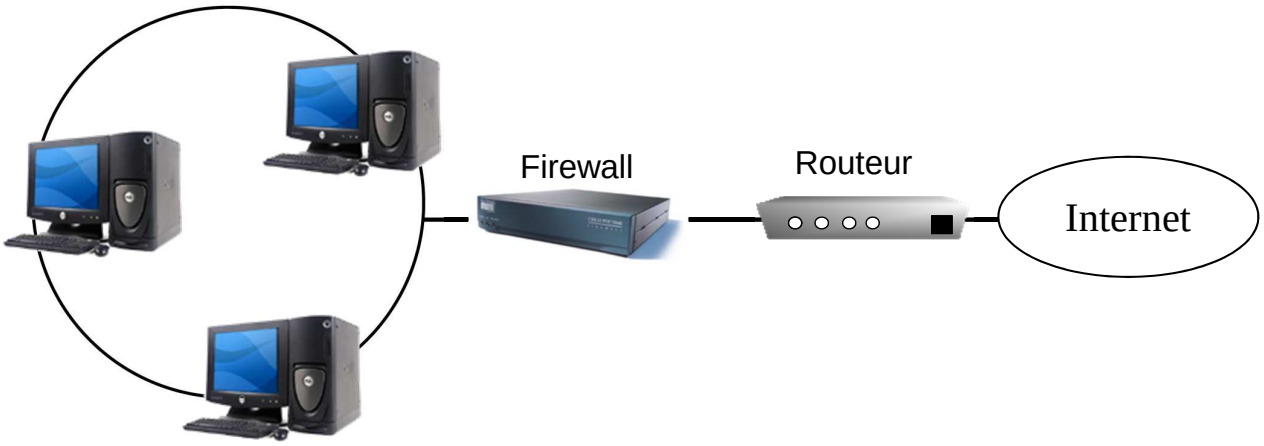


Architecture avec Firewall sans routeur

- On donne des adresses IP *privées* aux machines du réseau
  - Exemple : 10.1.1.1, 10.1.1.2
- Les serveurs ont *aussi* une adresse IP publique
  - Moyen: utilisation d'alias pour les cartes réseau
    - ✔ifconfig eth0 10.1.1.4
    - ✔ifconfig eth0:0 132.227.64.200
- Les clients ne peuvent pas dialoguer directement avec l'extérieur
- Passage par des **mandataires** internes
  - Ok pour certains services : smtp, nntp, web, ftp
- Plus compliqué ou impossible pour d'autres (sessions telnet)

Architecture avec Firewall et routeur

- Modèle avec Firewall et routage
- Le firewall est la seule machine visible à l'extérieur
  - ✔Le firewall effectue le contrôle d'accès
  - ✔Le routeur effectue le routage (translation d'adresse) ➔ NAT

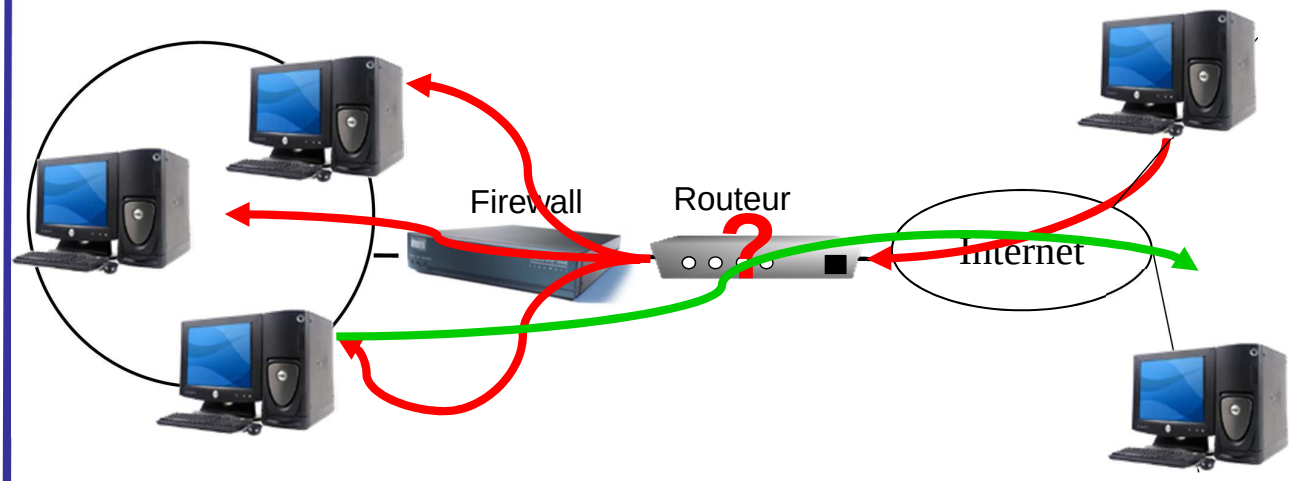


NAT

- Idée : le client fait passer ses communications par le routeur
  - Le routeur « déguise » les paquets pour faire croire qu'il en est l'émetteur
    - ✔Le serveur distant répond au routeur
    - ✔Le routeur fait suivre les réponses au client
  - C'est un exemple de NAT (Network Address Translation)
- Exemple :
  - Le poste 10.1.2.3 démarre une session telnet (TCP, port 23) en direction de 220.6.7.8
  - Le routeur remplace l'adresse d'origine (10.1.2.3) par sa propre adresse, et fait suivre à l'extérieur
  - Le site extérieur répond au routeur
  - Le routeur remplace l'adresse de destination (la sienne) par celle du demandeur 10.1.2.3
  - Le demandeur a obtenu sa réponse !

Pourquoi faire du NAT ?

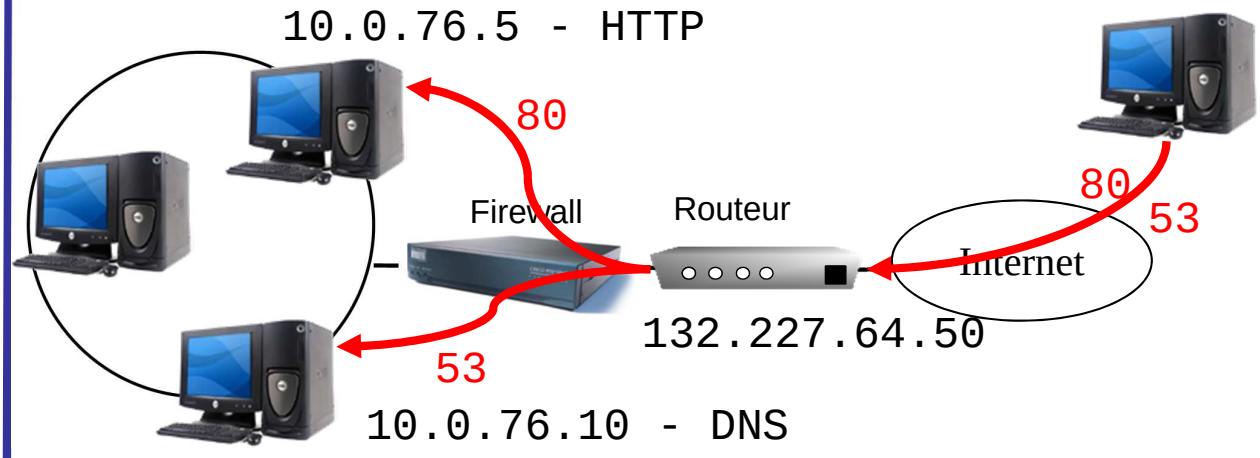
- Protéger ses machines clientes
  - Les connexions sortantes sont possibles
  - Les connexions entrantes sont interdites



Parefeu – NAT - SSL/TLS

Pourquoi faire du NAT ?

- Parfois, l'entreprise n'a qu'un nombre limité d'adresses IP
  - Elle veut déployer plusieurs services
  - Elle vaut faire du load-balancing
  - Le routeur oriente les paquets en fonction d'une politique précise

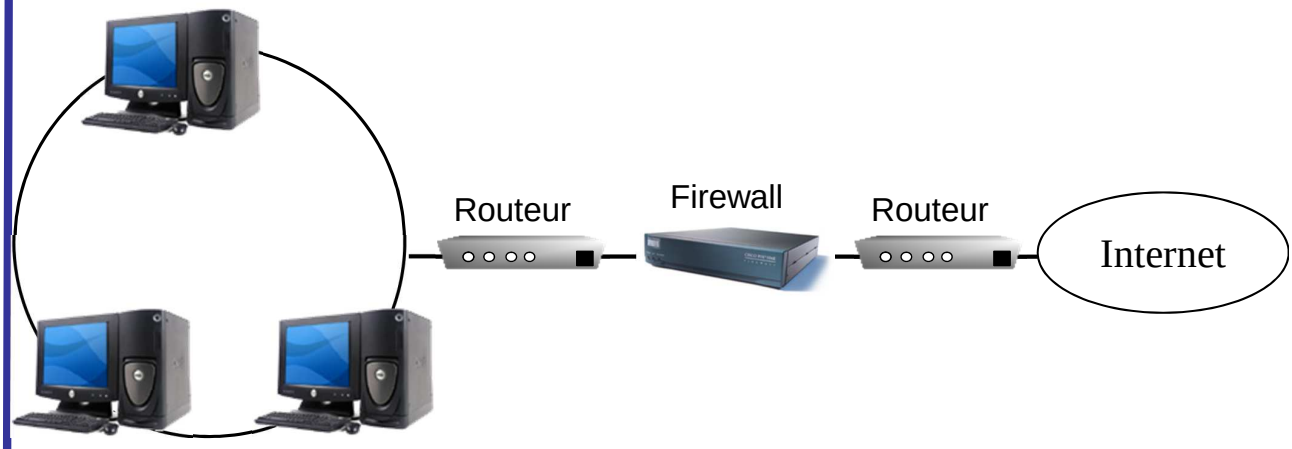


NAT n'est pas la solution ultime !

- Le NAT protège les machines clientes
  - Mais ils sont contournables !
- Méthodes :
  - Ping tunneling: Encapsulation des trames TCP dans les paquets ICMP
  - TCP traversing
  - Synchronous SYN
  - TCP/UDP Hole punching

Architecture avec Firewall et double routeurs

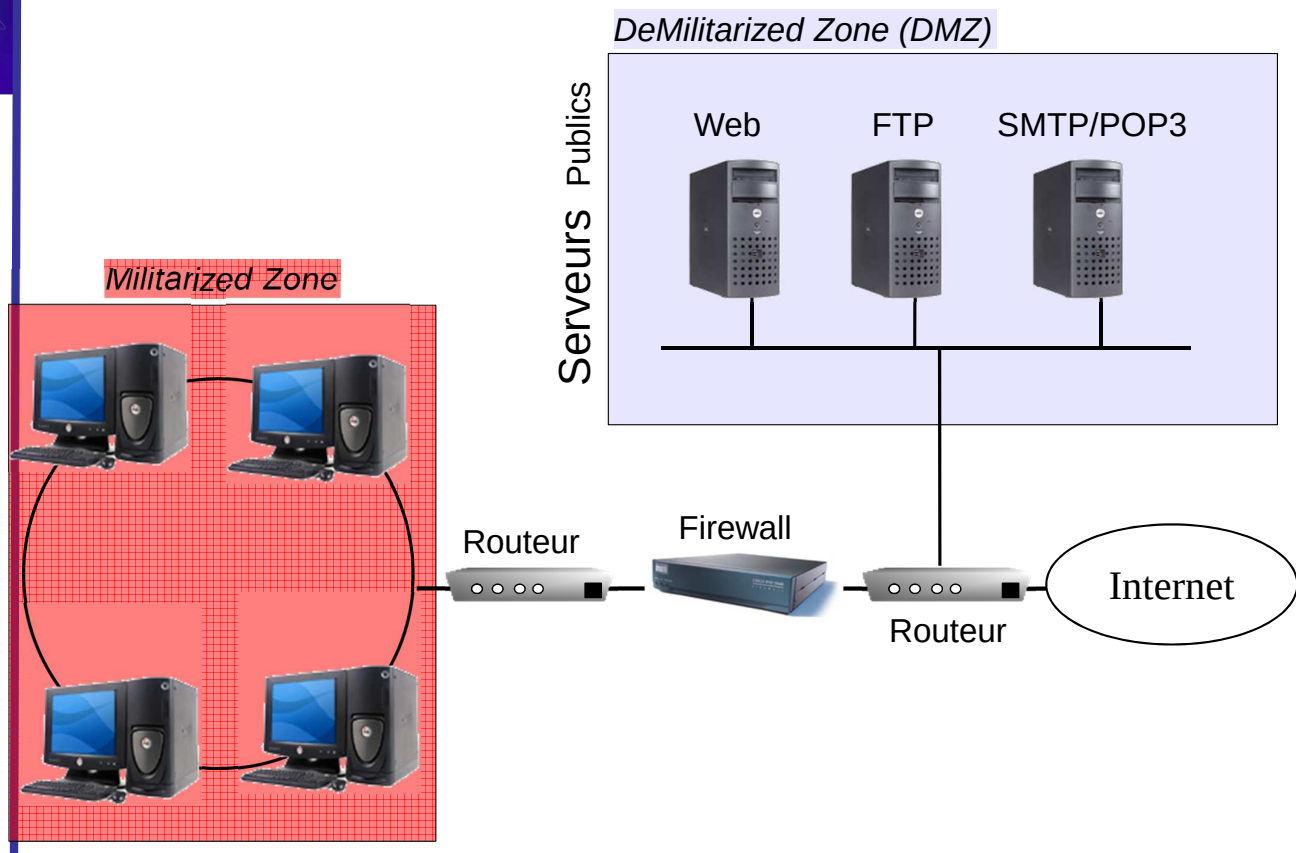
- Architecture à double routeur
  - Un routeur pour les connexions entrantes
  - Un routeur pour les connexions sortantes
  - Le firewall contrôle les accès entrants et sortants



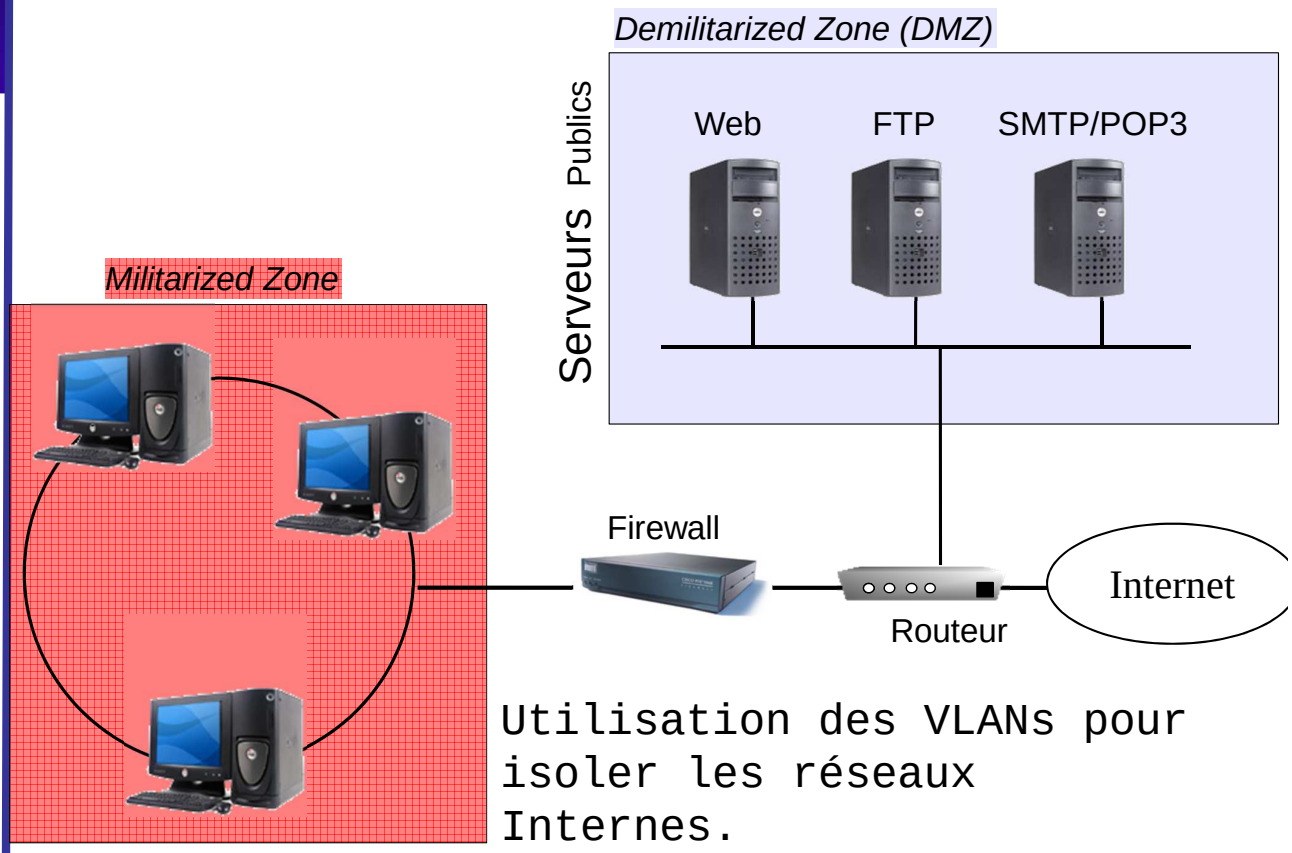
Firewall et zone démilitarisée (DMZ)

- ***Architecture à « DMZ »***
  - Découpage du réseau interne en 2 zones isolées
- Serveurs accessibles de l'extérieur, situés en « zone démilitarisée »
- Postes clients inaccessibles de l'extérieur (situé en « zone militarisée »)
- Variantes principales
  - Utilisation de deux routeurs
  - Utilisation d'un routeur « à 3 pattes »

Firewall et zone démilitarisée : 2 routeurs



Firewall et zone démilitarisée : Routeur à 3 pattes



Utilisation des VLANs pour isoler les réseaux Internes.

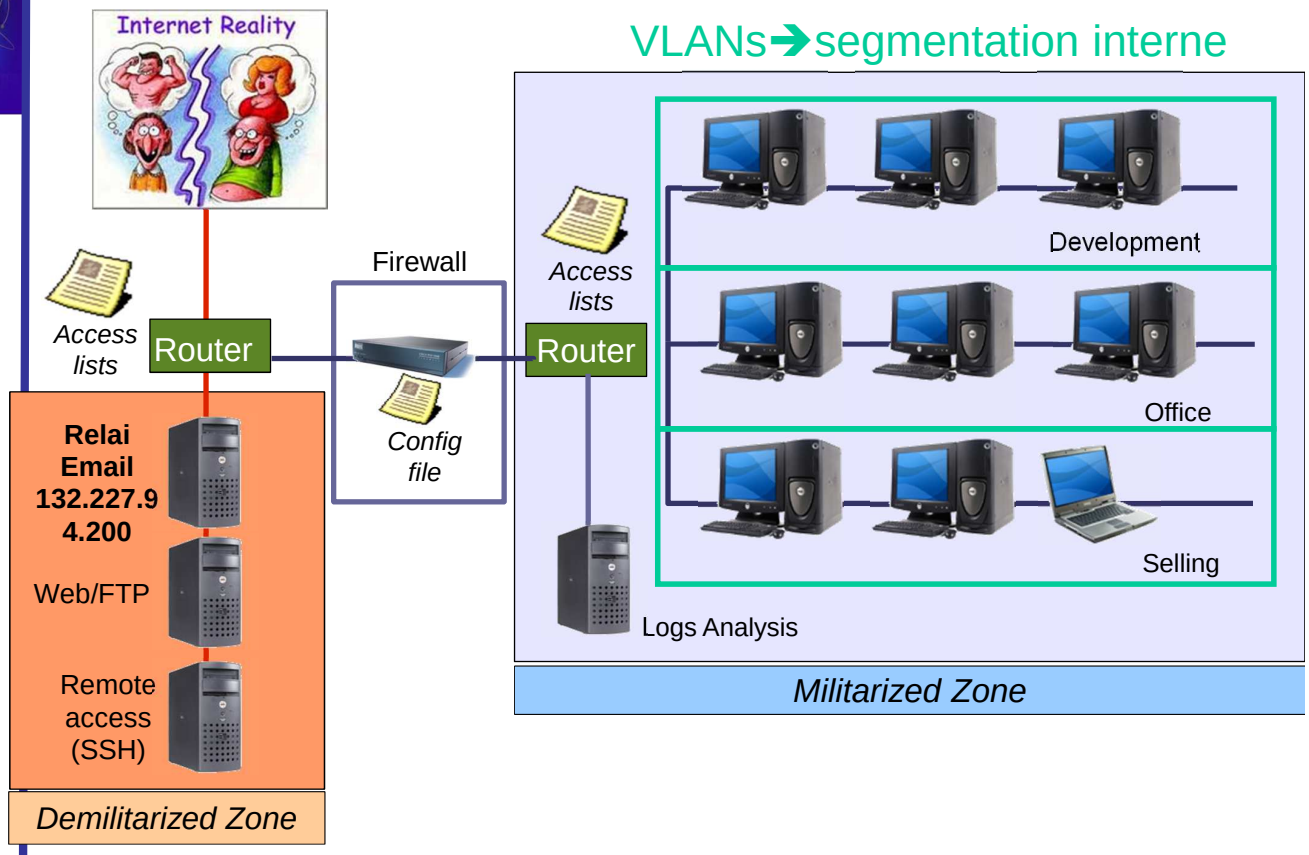
Firewall et zone démilitarisée

- ***Avantages-Inconvénients***
- Routeur à 3 pattes
  - plus économique, 1 seule machine à gérer
- 2 routeurs
  - Configuration plus simple
  - Meilleure sécurité
  - 2 étapes successives pour « braquer » le réseau interne

Utilisation des VLANs

- Le VLAN permet un regroupement d'un ensemble de machines
  - Permet l'isolement du groupe de machine
  - Regroupement par thème (pas par unité physique)
- Niveau de VLAN
  - Niveau 1 ➔ port physique sur le routeur
  - Niveau 2 ➔ adresse MAC de la machine
  - Niveau 3 ➔ sous-réseau IP ou port du service

Firewall, VLAN et zone démilitarisée



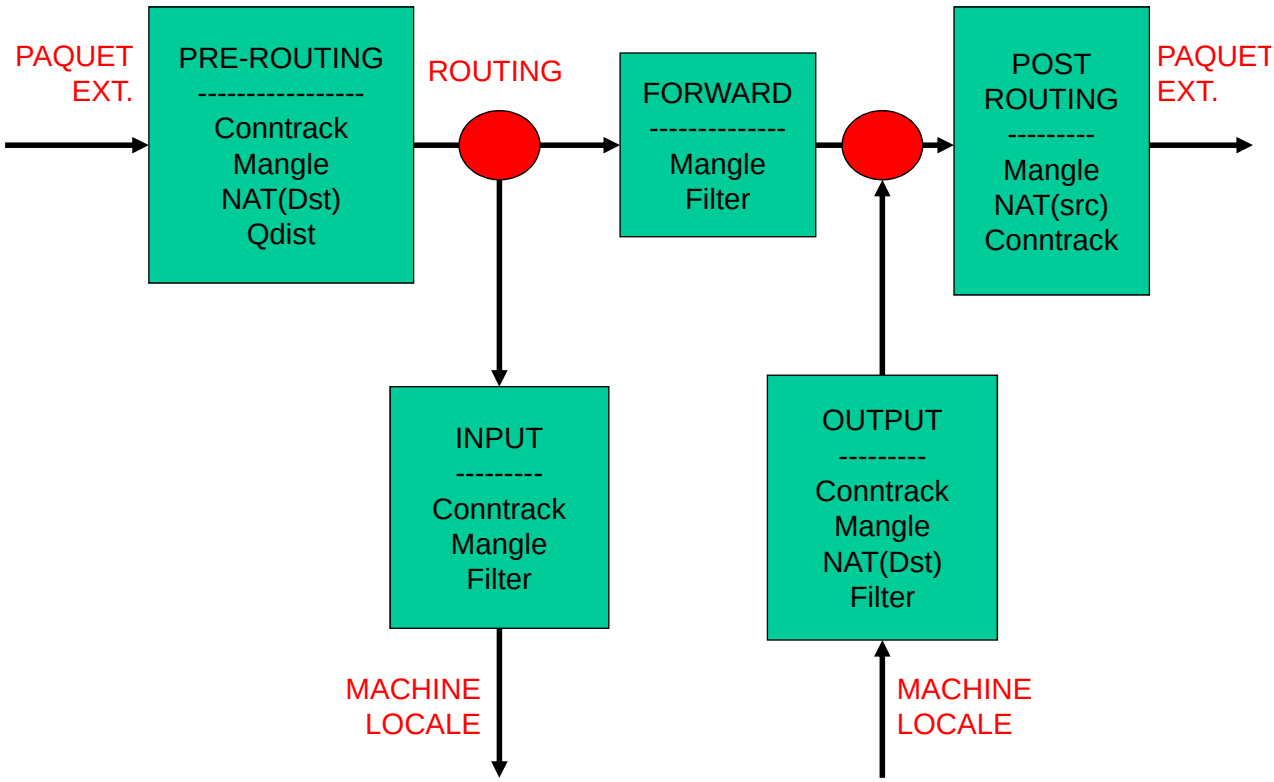
Format général et expressions des règles

- Principe : Des règles regroupées en chaînes
- Une chaine est
  - Un ensemble de règles
- Un règle est constituée
  - D'une cible ➔ en cas de concordance à qui dois-je faire suivre le paquet
  - De filtres sur l'adresse source [IP/Port] ➔ qui envoie le paquet
  - De filtres sur l'adresse destination [IP/Port] ➔ qui envoie le paquet
  - D'options (comme l'état de la connexion, l'utilisateur)
    - ✔Ex: La connexion a-t-elle été établie (ESTABLISHED) auparavant ?
- Applicable la grande majorité des firewalls
  - Iptables, ipfw, cisco ACL

Format général et expressions des règles

- Règles de filtrage
  - SMTP-ENTRANT-1 : Autoriser tous les paquets TCP entrants, @IP destination 132.227.94.200 port 25
  - SMTP-ENTRANT-2: Autoriser tous les paquets TCP provenant du port 25 de 132.227.94.200.
  - SMTP-SORTANT-1: Autoriser les paquets émis depuis 132.227.94.200 vers port 25 autre machine
  - SMTP-SORTANT-2: Autoriser les paquets émis depuis le port 25 d'une machine vers 132.227.94.200 et **ayant le bit ACK à 1**
- Problème sur les routeurs sans état :
  - La dernière règle n'empêche pas le passage de paquets «bidouillés» en direction du serveur de courrier
  - Laisse passer les tentatives d'attaques par saturation (DOS, déni de service)
- Solution : routeurs avec état
  - Le routeur mémorise les connexions TCP établies
  - Refuse les paquets qui n'en font pas partie, ou ne sont pas le début d'une nouvelle connexion autorisée

Iptables: Firewall + NAT !



Iptables: les types de table & les chaînes

- Types de table :
  - Conntrack ➔ Permet de gérer les connexions (iptable est statefull)
  - Mangle ➔ Permet la modification des options des paquets
  - Filter ➔ Permet le contrôle des paquets (sources, destinations)
  - Qdist ➔ pour faire de la QoS
  - NAT dst ➔ Changer l'adresse [IP/Port] de destination d'un paquet
  - NAT src ➔ Changer l'adresse [IP/Port] source d'un paquet
- Chaînes cibles préexistantes pour les tables :
  - INPUT, OUTPUT, FORWARD, PREROUTING, POSTROUTING (f° de la table)
  - ACCEPT ➔ Le paquet est accepté
  - DROP ➔ /dev/null
  - LOG ➔ Le paquet est tracé dans syslog [fabrique kernel]. Non bloquant !
  - REJECT ➔ Le paquet est rejeté et le firewall renvoie une erreur ICMP
  - RETURN ➔ Renvoie le paquet dans la chaîne précédente juste après l'endroit du branchement. Comportement par défaut à la fin d'une chaîne.
  - QUEUE ➔ Place le paquet dans l'environnement utilisateur

Iptables: manipulation des chaînes

- Création de la chaîne utilisateur « blacklist »

  iptables -N blacklist

- Suppression de la chaîne utilisateur « blacklist »

  iptables -X blacklist

- Vider une chaîne « chain » ou toutes les chaînes

  iptables -F [chain]

- Fixer le comportement par défaut de la chaîne « blacklist »

  iptables -P blacklist DROP

- Ajouter une règle « rule » à la chaîne « chain »

  iptables -A chain rule

- Insérer une règle « rule » à la chaîne « chain » après la position « num »

  iptables -I chain [num] rule

- Effacer une règle « rule » de la chaîne « chain » en position « num »

  iptables -D chain [num] [rule]

- Remplacer une règle « rule » en position « num » de la chaîne « chain »

  iptables -R chain [num] [rule]

Iptables : Un exemple de script

- Un firewall basique pour une machine cliente simple

  #Tout interdire en entrée par défaut

  Iptable –t filter –P INPUT DROP

  #Autoriser les connexions entrantes en loopback

  Iptables –t filter –A INPUT –i lo –s 127.0.0.0/8 –d 127.0.0.0/8 –j ACCEPT

  #ACCEPTE LES CONNEXIONS déjà ETABLIES

  Iptables –t filter –A INPUT –i lo -m state --state ESTABLISHED –j ACCEPT

  #ACCEPTE LES NOUVELLES CONNEXIONS ENTRANTES SUR LE PORT 80

  Iptables –t filter –A INPUT –i eth0 –m state --state NEW –p TCP --dport www –m limit --limit 1/s –j ACCEPT

  #ACCEPTE LE PING AVEC ANTIFLOOD

  Iptables –t filter –A INPUT –p icmp –icmp-type echo-request –m limit --limit 1/s –j ACCEPT

Iptables: Un exemple de script

- Ajout d'une black liste en entrée

 # créer une nouvelle chaîne

Iptable –N blacklist

 # fixer le comportement par défaut

Iptable –t filter –P blacklist RETURN

#On ajoute les adresses interdites

Iptables –t filter –A blacklist –i eth0 –s 10.0.0.0/8 –j DROP

Iptables –t filter –A blacklist –i eth0 –s 192.168.0.0/16 –j DROP

 # On ajoute un saut vers cette règle dans la règle INPUT du firewall

Iptables –t filter –I INPUT 3 –j blacklist

Iptables: Un exemple de script

#Ne pas faire suivre les paquets

Iptable –t filter –P FORWARD DROP

#Interdire toutes les connexions sortantes par défaut

Iptable –t filter –P OUTPUT DROP

#Autoriser les connexions sortantes en loopback

Iptables –t filter –A OUTPUT –o lo –s 127.0.0.0/8 –d 127.0.0.0/8 –j accept

#ACCEPTE LES CONNEXIONS déjà ETABLIES

Iptables –t filter –A OUTPUT –i lo -m state --state ESTABLISHED –j ACCEPT

#ACCEPTE LES NOUVELLES CONNEXIONS SORTANTES SUR LE PORT 22

Iptables –t filter –A OUPUT –m state --state NEW –p TCP –dport ssh -j ACCEPT

Iptables: les extensions

- Il existe de nombreux modules
  - « state » pour examiner l'état de la connexion
  - « string » pour examiner le contenu du paquet
  - « recent » pour gérer les derniers paquets reçus
  - « condition » pour déclencher un règle en fonction d'une variable dans /proc/net/ipt_condition/variable
  - « iplimit » pour limiter le nombre de connexions par @IP
  - « length » pour limiter le taille de certains paquets (ex: icmp)

Iptables : shorewall

- Il existe des scripts préfabriqués pour faire des firewalls « facilement »
- Beaucoup d'options prédéfinies !
- Voir les fichiers de configurations dans /etc/shorewall et leurs commentaires.
- Efficace en terme de temps de déploiement et de sécurité !
- Fichier /etc/shorewall/interfaces
  - Choix des interfaces et des options/zones associées
- Fichier /etc/shorewall/rules

  ACCEPT net fw icmp 8 - - 3/sec:10

  ACCEPT net fw tcp 22 -
- Lancement : **/etc/init.d/shorewall restart**
- Génération automatique des règles iptables. Pour les voir:

  /etc/init.d/shorewall status

Parefeu – NAT - SSL/TLS

Iptables: Exemple de code Shorewall

```
Chain INPUT (policy DROP 0 packets, 0 bytes)
target     prot opt in     out     source               destination
ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0
DROP       !icmp --  *      *       0.0.0.0/0            0.0.0.0/0 state INVALID
eth0_in    all  --  eth0   *       0.0.0.0/0            0.0.0.0/0
Reject     all  --  *      *       0.0.0.0/0            0.0.0.0/0
LOG        all  --  *      *       0.0.0.0/0            0.0.0.0/0 ...
                                    ... LOG flags 0 level 6 prefix `Shorewall:INPUT:REJECT:'
reject     all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain eth0_in (1 references)
target     prot opt in     out     source               destination
dynamic    all  --  *      *       0.0.0.0/0            0.0.0.0/0 state INVALID,NEW
net2fw     all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain net2fw (1 references)
target     prot opt in     out     source               destination
ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0 state RELATED,ESTABLISHED
ACCEPT     udp  --  *      *       132.227.64.37        0.0.0.0/0
ACCEPT     tcp  --  *      *       132.227.64.37        0.0.0.0/0
ACCEPT     udp  --  *      *       224.0.1.1            0.0.0.0/0 udp dpt:123
ACCEPT     tcp  --  *      *       132.227.64.0/24      0.0.0.0/0 tcp dpt:111
ACCEPT     tcp  --  *      *       132.227.64.0/24      0.0.0.0/0 tcp dpts:6000:6010
ACCEPT     udp  --  *      *       132.227.64.0/24      0.0.0.0/0 udp dpts:6000:6010
ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0 tcp dpts:63000:64000
ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0 tcp dpt:22
ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0    ...
                                      ... icmp type 8 limit: avg 3/sec burst 10
net2all    all  --  *      *       0.0.0.0/0            0.0.0.0/0
```


Parefeu – NAT - SSL/TLS

Iptables: Voyage d'un paquet

TCP SYN de 132.227.64.8 port 22

```
Chain INPUT (policy DROP 0 packets, 0 bytes)
target     prot opt in     out     source               destination
ACCEPT     all  --  lo     *       0.0.0.0/0            0.0.0.0/0
DROP       !icmp --  *      *       0.0.0.0/0            0.0.0.0/0 state INVALID
eth0_in    all  --  eth0   *       0.0.0.0/0            0.0.0.0/0
Reject     all  --  *      *       0.0.0.0/0            0.0.0.0/0
LOG        all  --  *      *       0.0.0.0/0            0.0.0.0/0 ...
                                    ... LOG flags 0 level 6 prefix `Shorewall:INPUT:REJECT:'
reject     all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain eth0_in (1 references)
target     prot opt in     out     source               destination
dynamic    all  --  *      *       0.0.0.0/0            0.0.0.0/0 state INVALID,NEW
net2fw     all  --  *      *       0.0.0.0/0            0.0.0.0/0

Chain net2fw (1 references)
target     prot opt in     out     source               destination
ACCEPT     all  --  *      *       0.0.0.0/0            0.0.0.0/0 state RELATED,ESTABLISHED
ACCEPT     tcp  --  *      *       0.0.0.0/0            0.0.0.0/0 tcp dpt:22
ACCEPT     tcp  --  *      *       132.227.64.37        0.0.0.0/0
ACCEPT     tcp  --  *      *       132.227.64.0/24      0.0.0.0/0 tcp dpt:111
ACCEPT     tcp  --  *      *       132.227.64.0/24      0.0.0.0/0 tcp dpts:6000:6010
ACCEPT     udp  --  *      *       132.227.64.0/24      0.0.0.0/0 udp dpts:6000:6010
ACCEPT     icmp --  *      *       0.0.0.0/0            0.0.0.0/0    ...
                                      ... icmp type 8 limit: avg 3/sec burst 10

Chain dynamic (1 references)
… vide … ➔ return
```

Match ? Non
Match ? Non
Match ? Oui
Match ? Oui
Match ? Non
Match ? Oui

Paquet ACCEPTE

DHCP

La base du réseau

Routage IP et couche liaison

Couche Transport : TCP/UDP

Configuration réseau

Outils réseau

Parefeu – NAT

DHCP

DNS

IDS et Analyse

Rôle du service DHCP

- Le service DHCP (Dynamic Host Configuration Protocol)
- Il utilise le port 67 (serveur) et 68 (client)
- Le nom de machine et l'IP sont fournit par le service DHCP
  - Utilise les adresses ethernet (MAC) des cartes réseaux
  - La machine diffuse sa demande (car le serveur est inconnu)
- En cas de "timeout" de la requête
  - Comportement 1: Une adresse par défaut est prise par la machine
  - Comportement 2 : Le réseau ne démarre pas
- Les informations suivantes sont aussi envoyées par le serveur DHCP :
  - Masque de réseau (Netmask) ➔ Adresse de réseau (network adresse)
  - Adresse de diffusion (Broadcast address)
  - Adresse de la passerelle de sortie (Gateway)
  - Adresse des serveurs de noms (DNS Servers)
  - Nom du domaine d'authentification NIS (NIS domain)
  - …

DHCP

Utilité du DHCP

- Permet une gestion centralisée des adresses IP
  - Utilise les @ MAC du côté du serveur
  - Le serveur peut assigner des @IP fixe en fonction de l'adresse MAC du client
  - Le serveur peut assigner des @IP dynamiques temporaires
  - Une même adresse peut être utilisée pour désigner plusieurs machines au cours du temps
  - Il n'est pas nécessaire d'avoir autant d'adresses que d'abonnés si tous les abonnés ne se connectent pas en même temps
- Permet de changer facilement la configuration réseau d'une machine
- Permet l'installation ou le démarrage de machine par le réseau (BOOTP)
- Permet la gestion des machines sédentaires et mobiles
- DHCP souvent considéré comme une faille de sécurité
  - Configuration automatique sur des clients inconnus

DHCP

Le protocole DHCP

- RFC 951 (Bootp) , 1542, 2131 (dhcp), 2132
- Les requêtes et les messages DHCP
  - **DHCPDISCOVER:** envoyé par le client pour localiser les serveurs DHCP disponibles
  - **DHCPOFFER:** réponse du serveur à un paquet DHCPDISCOVER, qui contient les premiers paramètres (en particulier l'adresse IP du serveur)
  - **DHCPREQUEST:** contient les requêtes diverses du client (ex: prolongation d'un bail)
  - **DHCPACK:** réponse du serveur qui contient des paramètres et **l'adresse IP** du client
  - **DHCPNAK**: réponse du serveur pour signaler au client un refus
  - **DHCPDECLINE:** le client annonce au serveur que l'adresse est déjà utilisée
  - **DHCPRELEASE:** le client libère son adresse IP
  - **DHCPINFORM:** le client demande des paramètres locaux, il a déjà son adresse IP

Le protocole DHCP

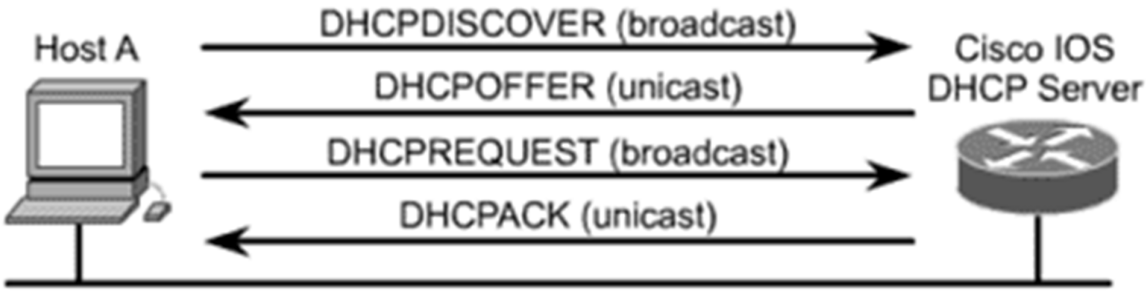
Le premier paquet émis par le client est un paquet de type DHCPDISCOVER.
- Il est envoyé « à tout le monde » (broadcast, 255.255.255.255)
- Il contient l'adresse IP 0.0.0.0. On utilise l'adresse de liaison (@MAC)

Le serveur répond par un paquet DHCPOFFER avec son IP
- Soit en utilisant l'adresse MAC de celui qui a émis le paquet
- Soit en répondant « à tout le monde » (broadcast, 255.255.255.255)

Le client fait un DHCPREQUEST pour obtenir son IP
- Utilisation de l'IP contenue dans le premier DHCPOFFER reçu
- Envoyer « à tout le monde » pour avertir tous les serveur DHCP

Le serveur répond simplement par un DHCPACK avec l'adresse IP pour confirmation de l'attribution
- Il y a vérification, par le serveur, de l'IP qui va être attribuée en utilisant le protocole ICMP « Echo Request »
- En cas de duplication , le serveur envoie DHCPDECLINE, et on recommence

DHCP: Complément



- DHCP est un protocole assez bas niveau.
- Toute adresse IP délivrée par un serveur DHCP a une durée de vie appelée « bail ».
- A la moitié de la durée du bail, un client doit renouveler son bail.
  - Le client peut en faire la demande en envoyant un DHCPREQUEST.
- Le serveur vérifie la présence du client à la fin du bail en envoyant un DHCPNACK.
  - En cas de non réponse l'adresse est libérée pour ré-utilisation

DHCP

Configuration du serveur ISC DHCPd

- Configuration par le fichier « */etc/dhcpd.conf* »

```
#options globales
deny unknown-clients; # interdire les clients inconnus
deny client-updates; #interdit les demande de mise à jour émise par les clients
deny bootp; #interdire le protocole bootp

#Configuration des options pour un sous-réseau spécifique (ici le 132.227.64.0/24)
subnet 132.227.64.0 netmask 255.255.255.0 {

        authoritative; # le serveur fait autorité sur le sous-réseau
        option routers 132.227.64.15; # la passerelle de sortie pour accéder à internet
        option subnet-mask 255.255.255.0; # masque pour le sous-réseau local

        option domain-name "lip6.fr"; #domaine de ce sous-réseau (devrait appartenir au DNS)
        option domain-name-servers 132.227.64.13, 132.227.60.30, 132.227.60.2; # les DNS

        allow bootp; #autorise le protocole bootp sur ce sous-réseau (annule la directive locale)
        allow booting; # autorise le boot par le réseau (boot pxe)
        deny unknown-clients; #interdire les clients inconnus (doublon avec l'option globale)

        default-lease-time 21600; # temps du bail par défaut (en secondes)
        max-lease-time 43200; # temps maximum du bail (en secondes)
        range 132.227.64.240 132.227.64.250; # IP réservé au pool DHCP

        # déclare une machine SANS lui attribuer une IP fixe
        host PORTABLE_DENIS {
                hardware ethernet 00:04:75:96:9D:F8;
        }

       # déclare une machine en lui attribuer une IP fixe et un nom
        host PORTABLE_MANU {
                hardware ethernet 00:04:75:96:AA:CD;
                fixed-address 132.227.64.2;
                option host-name "pmanu";
        }
}
```


DHCP

Redondance DHCP

- Si vous choisissez DHCP comme service d'attribution d'IP pour toutes vos machines, il devient critique !!
- Le DHCP n'est pas fait pour les serveurs
- Si vous déployez une solution basée totalement sur DHCP, en cas de crash ➔ plus de réseau
- Relai DHCP entre des sous-réseaux : « ***dhcrelay*** »
- Redondance possible pour plus de sûreté. On ajoute au fichier « */etc/dhcpd.conf* » :

```
failover peer "eros" {
        primary;
        #THIS PRIMARY DHCP SERVER (132.227.64.25=eros.lip6.fr)
        address 132.227.64.25;
        port 520; # port d'écoute pour l'échange d'informations entre serveurs
        #PEER SLAVE DHCP SERVER (132.227.64.26=no dns entry)
        peer address 132.227.64.26;
        peer port 519; # port d'écoute pour l'échange d'informations entre serveurs
        #nombre de secondes avant que l'autre hôte ne prenne le relai
        load-balance-max-seconds 3;
}
```

DNS

La base du réseau

Routage IP et couche liaison

Couche Transport : TCP/UDP

Configuration réseau

Outils réseau

Parefeu – NAT

DHCP

DNS

IDS et Analyse

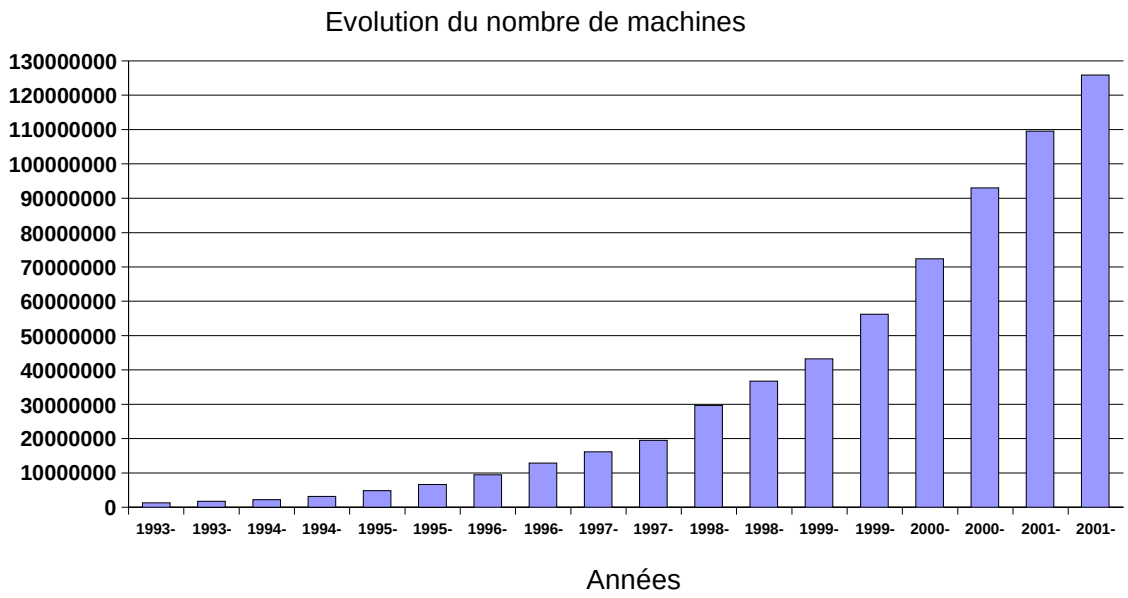
DNS

- Les usagers préfèrent utiliser les noms logiques !
- Exemples :
  - Adresse courrier (**legond@src.lip6.fr**) plutôt qu'une adresse IP (legond@[132.227.64.100])
  - Nom de site web (**http://www.hardware.fr**) plutôt qu'une URL IP (**http://83.243.20.80**)
- ➔ Besoin d'un ensemble de mécanismes de création, d'administration, de mise en relation pour des noms logiques, des adresses, des attributs.
- Au début de l'Internet, les noms étaient définis **localement** sur chaque hôte dans un **fichier (*/etc/hosts*** en UNIX).
- Mise à jour de « ***/etc/hosts*** » par ftp la nuit **automatiquement** ou **manuellement** à partir d'une version référence pour suivre l'évolution du réseau Internet

DNS

Historique

- Problème : Le nombre de machines référencés a explosé !



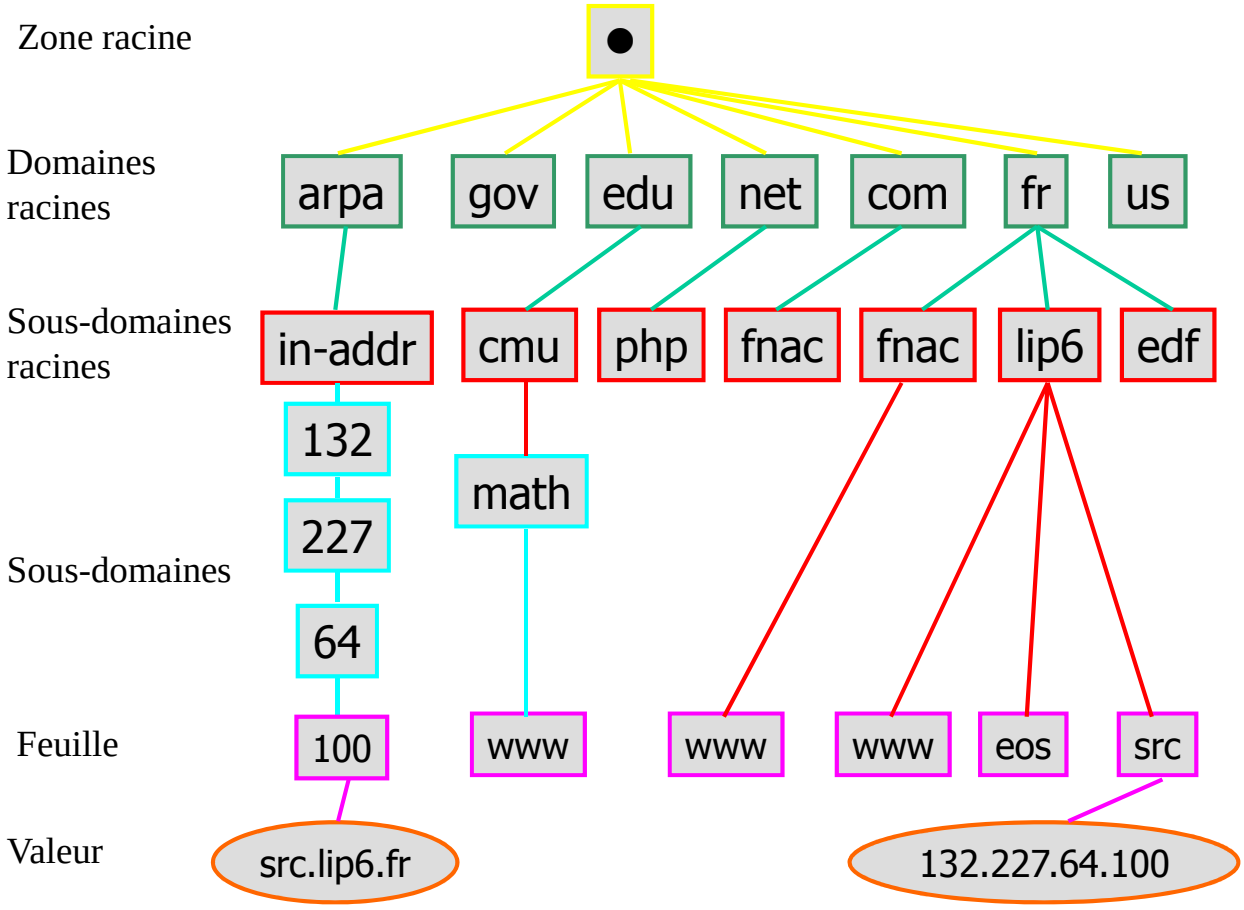
DNS

Rôle du service DNS

- **1984** : Création d'un service d'annuaire distribué (base de données distribuée d'informations)
- **DNS = « Domain Name Service »**
  - Service le plus important et le plus utilisé d'Internet.
  - Tout le réseau (quasiment) repose sur ce service contrôlé par les USA
  - RFC de base: 1034, 1035
  - Nombreuses RFC (voir www.dns.net/dnsrd/rfc)
- Un des protocole de base de l'internet . Fonctions principales:
  - **C'EST UN ANNUAIRE DE MACHINES**
  - DNS permet d'associer un nom « humain » à une adresse IP et vice-versa
- Spécificités du protocole DNS
  - Le DNS offre un identifiant textuel unique !
  - Accessible au moyen d'un espace de nommage hiérarchique unifié
  - De même qu'il existe une adresse réseau et une adresse machine, il existe des noms de domaines et des noms de machines

DNS

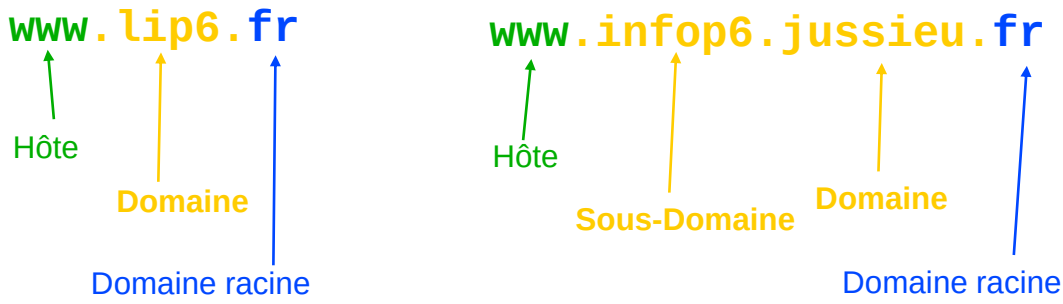
Structure hiérarchique du DNS



DNS

Structure hiérarchique du DNS

– Exemples :



– Au niveau le plus haut: Plusieurs centaines de noms de domaines.
  - ✔com : Noms génériques de domaines.
  - ✔fr : Noms géographiques de domaines.
  - ✔arpa : Correspondance adresses IP vers noms.
– Aux niveaux intermédiaires: Des noms de domaines (qui sont des sous-domaines).
– Au niveau des feuilles: Des sous-domaines composés d'hôtes ou définissant des services.

Top Level Domains (TLD) DNS

DNS

- **Des domaines connus :**
  - **.com** : Organismes commerciaux (Verisign)
  - **.net** : Prestataires réseaux (Verisign)
  - **.org** : Autres organisations (Verisign)
  - **.info** : Orgs d'information (Afilias Limited)
  - **.« Code pays »** : entreprises et services d'un pays (avec le nouveau **.eu**)
  - **.edu** : Institutions d'éducation US
  - **.gov** : Organisations gouvernementales US
- **Des domaines exotiques :**
  - **.aero** : Industries aéronautiques (SITA)
  - **.biz** : Affaires (NeuLevel, Inc).
  - **.coop** : Associations cooperative (Dot Cooperation LLC).
  - **.museum** : Musées (Museum Domain Management Association).
  - **.name** : Individus (Global Name Registry).
  - **.mil** : Armée US
  - **.int** : Organisations internationales
- ……..

Gestion des DNS

DNS

- Désignation dans l'arborescence
  - Les feuilles désignent une machine (eos, scylla)
  - La machine peut porter le nom d'un service (ftp, dns, …)
  - Les nœuds intermédiaires sont composés d'un ensemble de ressources
  - L'organisation gérant un domaine peut déléguer la gestion d'un sous-domaine.
  - Pour créer un sous-domaine, il faut donc avoir l'autorisation de l'organisme gérant le domaine père.
  - **IMPORTANT: l'organisation DNS est différente de la topologie IP sous jacente !**

DNS

Gestion des DNS

- En plus de la dé-corrélation IP / nom de domaine, il y a dé-corrélation entre les unités d'administration des domaines (zone) et les domaines
- Un domaine est :
  - un ensemble de noms qui ont un même suffixe
  - **un découpage syntaxique** de l'espace de nommage Internet
- Une zone est :
  - une unité d'administration (tous les membres d'une zone sont servis par un même serveur)
  - une zone regroupe **un ensemble de domaines voisins** qui ne se recouvrent pas.
  - **un découpage administratif** définissant **la portée d'action** des serveurs de noms (suivant les délégations internes de gestions)
  - Ex: lip6.fr, src.lip6.fr

DNS

Désignation et recherches

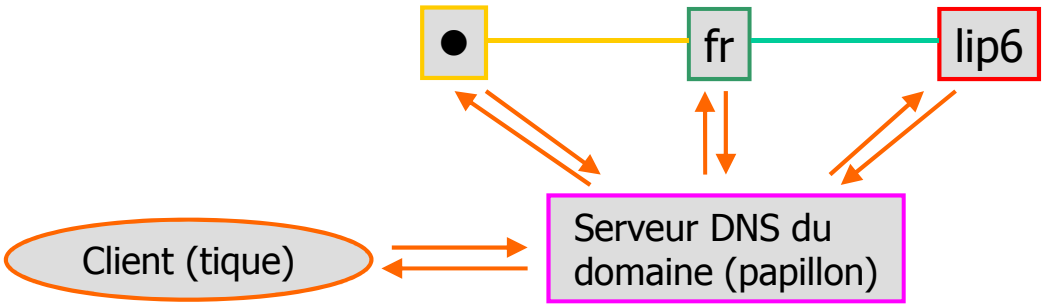
- FQDN : Fully Qualified Domain Name
  - Construit en suivant le chemin de la racine aux feuilles
  - On sépare les intermédiaires par des « . »
  - Ex: *www.java.sun.com*, *www.infop6.jussieu.fr*
- Nom « non-fqdn » : ce sont des noms relatifs qui sont recherchés dans le domaine courant
  - Ex: *www*, *www.ufr-info-p6*
- La recherche :
  - se fait par défaut dans le domaine auquel appartient le serveur
  - est étendu au domaine « . » en cas d'échec
  - peut être forcée à partir du répertoire racine en ajoutant un « . » à la fin de la chaîne cherchée

DNS

Modes de recherche

- Un indicateur dans la requête décrit la façon de la traiter : **itérative ou récursive.**
- Sur chaque machine existe une liste de 13 serveurs racines (fichier ***/var/named/named.ca***)
- Mode itératif
  - On **interroge successivement** les serveurs

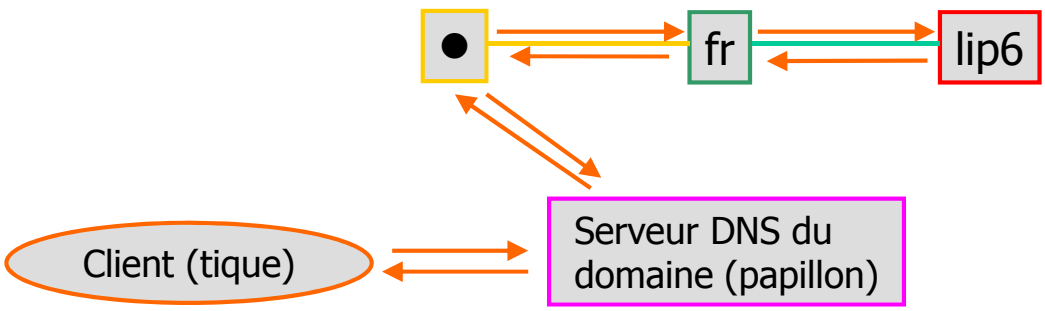
```
tique.infop6.jussieu.fr> nslookup eos.lip6.fr
Server:         134.157.116.123
Address:        134.157.116.123#53
Non-authoritative answer:
Name:   eos.lip6.fr
Address: 132.227.64.45
```



DNS

Modes de recherche

- Mode récursif
  - Chaque serveur visité prend l'initiative **d'interroger le serveur suivant** pour obtenir pour lui même la réponse à la question posée
  - La réponse revient en **visitant tous les sites**
  - On note la résolution effectuée dans les **caches** de tous les serveurs visités



DNS

Type de requête

- A: demande d'une adresse de machine
- CNAME: demande le nom réel (canonique) pour un alias.
- PTR: le nom de machine de l'adresse IP
- MX: les serveurs de mail (envoie)
- NS: le(s) serveur(s) DNS gestionnaire(s) du domaine
- SOA: des informations sur le domaine (« start-of-authority »)
- HINFO: demande le CPU et l'OS du serveur (optionnel et dangereux)
- TXT: informations textuelles sur le domaine
- Autres informations: MINFO, UINFO, WKS,ANY, AXFR, MB, MD, MF, NULL

DNS

Outils DNS: dig

- « ***dig [@server] [options dig] [nom] [type] [classe] [options requête]***» (très verbeux, voir les options)

```
legond@tique.infop6.jussieu.fr> dig  www.efrei.Fr +short
efrei.opixido.com.
62.4.72.6

legond@tique.infop6.jussieu.fr> dig -x 62.4.72.6 +short
62.4.72.6.not.updated.above.net.

legond@tique.infop6.jussieu.fr> dig efrei.Fr a +short
194.2.204.17

legond@tique.infop6.jussieu.fr> dig efrei.Fr ns
...
;; ANSWER SECTION:
efrei.fr.            86377   IN      NS      cerbere.efrei.fr.
efrei.fr.            86377   IN      NS      turner.efrei.fr.
;; ADDITIONAL SECTION:
cerbere.efrei.fr.       86377   IN      A       194.2.204.4
...
```

DNS

Outils DNS: nslookup

- « ***nslookup [-option ...] [host-to-find | -[server]]*** »
  - ✔Peut être lancé comme un shell
  - ✔Peut être lancé en ligne de commande

```
legond@tique.infop6.jussieu.fr>  nslookup www.epita.fr
Non-authoritative answer:
Name:    www.epita.fr
Address: 163.5.254.17

legond@tique.infop6.jussieu.fr> nslookup -query=ns epita.fr
Non-authoritative answer:
epita.fr        mail exchanger = 65 smtp-relay.epita.fr.
epita.fr        mail exchanger = 40 smtp1.epita.fr.
epita.fr        mail exchanger = 40 smtp2.epita.fr.

legond@tique.infop6.jussieu.fr> nslookup
>set type=soa
Non-authoritative answer:
epita.fr
        origin = ns1.epi.net
        mail addr = postmaster.epita.fr
        serial = 2005072002
        refresh = 21600
        retry = 3600
        expire = 604800
        minimum = 86400
```


DNS

Outils DNS: whois

- Obtenir des informations sur les propriétaires et les gestionnaires d'un domaine
- Online: www.ripe.net, www.arin.net, www.afrin.net
- « ***whois*** » a de nombreuses options et des nombreuses possibilités. Lire le manuel !
- Whois sur une @IP donne des informations sur le propriétaire de la classe d'IP! (l'hébergeur)
- Whois sur un domaine donne des informations sur le propriétaire du domaine !
- Essayez « ***whois liberation.fr*** » et « ***whois 80.15.238.13*** »

DNS

Outils DNS: whois

```
tique.infop6.jussieu.fr> whois epita.fr
domain:       epita.fr
address:      Ecole Pour l'Informatique et les Techniques Avancees
address:      14-16, rue Voltaire
address:      94270 Le Kremlin-Bicêtre
address:      FR
admin-c:      NS1297-FRNIC
tech-c:       JB371-FRNIC
zone-c:       NFC1-FRNIC
nserver:      ns1.epi.net
nserver:      ns2.epi.net
nserver:      joe.hittite.isp.9tel.net
mnt-by:       FR-NIC-MNT
mnt-lower:    FR-NIC-MNT
changed:      nic@nic.fr 20050124
source:       FRNIC

person:       Nicolas Sadirac
address:      Ecole Pour l'Informatique et les Techniques Avancees
address:      106-112, boulevard de l'Hopital
address:      75013 Paris
address:      FR
....
```


DNS

Les performances: Répartition et Distribution

- Un serveur centralisé ne pourrait pas supporter les requêtes
- Une des meilleurs réussite en terme de répartition de charge
- Pour rappel : Sur chaque machine (serveur et cliente) existe une liste de 13 serveurs racines (fichier ***/var/named/named.ca***)
- La majorité des requêtes sont locales (sur un sous-réseau et/ou sous-domaine)
  - On utilises ses propres serveurs
  - On allège les autres serveurs !
- Répartition de charge grâce au DNS : le DNS peut renvoyer plusieurs IP différentes pour un même nom !

DNS

Les Performance: cache DNS

- Gestion d'un cache local et entre serveurs
  - Prise en compte d'un délai de péremption des données (information expire du SOA)
  - Raccourci les délais de réponse et économise la bande passante
  - Délai de prise en compte d'une modification d'une entrée DNS
  - Cohérence faible ➔ délai de MAJ (~3 jours)
- Lors d'une réponse:
  - Si elle vient du cache, elle est non fiable (« non authoritative »)
  - Si elle vient d'un serveur dit « authoritative », elle est fiable.
- Le temps de résidence dans le cache est un paramètre important.
- La non fiabilité des réponses peut poser des problèmes:
  - Pour atteindre certains serveurs
  - Pour la sécurité

DNS

DNS: Sûreté de fonctionnement

- DNS est un service critique !
  - Obligation d'avoir au moins deux serveurs DNS
  - 2 machines physiques différentes doivent assurer le DNS
  - Problème de synchronisation entre serveurs
- Mise en place d'un protocole de cohérence
  - Synchronisation entre serveur
  - Le secondaire (esclave) se synchronise sur le primaire (maître)
- Les DNS secondaires font des transferts de zone:
  - Lors de leur lancement
  - A intervalle régulier pour se synchroniser
- Si le DNS primaire est non accessible
  - On démarre avec une copie locale des zones
  - On se synchronise dès le retour du serveur primaire
- Le numéro de série indique la date des changements
- ATTENTION A LA SECURITE !!!

DNS

Configuration DNS (serveur ISC Bind)

- Les fichiers de configuration sont
  - Configuration des clients « ***/etc/resolv.conf*** »
    - ✔ *search lip6.fr #domaine(s) de recherches*
    - ✔ *domain lip6.fr #domaine local*
    - ✔ *order local,bind #ordre de résolution: local (/etc/hosts) puis dns*
    - ✔ *nameserver 132.227.64.13*
    - ✔ *nameserver 132.227.60.30*
    - ✔ *nameserver 132.227.60.2*
  - Configuration du service: « ***/etc/named.conf*** » (intègre maintenant « ***/etc/named.boot*** »)
  - Informations sur les zones administrées dans le répertoire « ***/var/named/*** »
  - Vérification des configurations: « ***dnswalk*** »

DNS

« /etc/named.conf »

- Sécurité:
  - Gestion des clefs pour l'authentifications des pairs:
    - ✔ Section « ***key domaine_name*** » pour définir sa clef
    - ✔ Section « ***trusted-keys*** » pour définir les pairs de confiance !
  - Section « ***controls*** » permet le contrôle d'accès au serveur
  - Utilisation des ACL pour le contrôle d'accès
- Section « ***logging*** » permet de définir les traces
- Section « ***options*** » pour définir les options du serveur
- Sections « ***zone "nom"*** » pour chaque zone que gère le serveur
  - Contient essentiellement le nom des fichiers de zone
  - Le type de la zone (master, slave, stub [NS slave])
  - Comment se fait la MAJ

ISC BIND: Les fichiers de zone

- Association noms vers IP

```
$ORIGIN example.com ; origine du fichier (base ajoutée à toutes les entrées)

$TTL 86400 ; Valeur de la durée de validité des informations de la zone (ici 1 jour)

; début de la description de la zone (entrée SOA)
; nom_zone (@=valeur de $ORIGIN) IN SOA nom-serveur-fqdn. email-responsable. (
@     IN     SOA    dns1.example.com.      hostmaster.example.com. (
                    2001062501 ; n° de série (doit être incrémenté après  chaque MAJ du fichier)
                    21600      ; Délai après lequel le serveur esclave doit se mettre à jour (ici 6 heures)
                    3600       ; Délai entre chaque essai de synchro du serveur esclave (ici 1h)
                    604800     ; Délai à partir duquel un serveur esclave arrête de répondre
                    86400 )    ; doublon avec $TTL

;Entrées DNS de la zone (machines et sous-domaines)
          IN     NS     dns1.example.com. ; les serveurs de noms (champs NS)
          IN     NS     dns2.example.com.
          IN     MX     10     mail.example.com. ; les serveurs de mails 10 et 20 sont des priorités
          IN     MX     20     mail2.example.com.
          IN     A      10.0.1.5          ; entrée par défaut du domaine (ex: ping example.com)
server1   IN     A      10.0.1.5          ; déclaration de la machine server1 sur le domaine
server2   IN     A      10.0.1.7          ; déclaration de la machine server2 sur le domaine
dns1      IN     A      10.0.1.2          ; déclaration de la machine dns1 sur le domaine
dns2      IN     A      10.0.1.3          ; déclaration de la machine dns2 sur le domaine
ftp       IN     CNAME  server1           ; déclaration d'un alias (ftp=server1)
mail      IN     CNAME  server1           ; déclaration d'un alias (mail=server1)
mail2     IN     CNAME  server2           ; déclaration d'un alias (mail2=server2)
www       IN     CNAME  server2           ; déclaration d'un alias (www=server2, pas l'hôte par défaut)
```


ISC BIND: Les fichiers de zone

- Association IPs vers noms (reverse)

```
$ORIGIN 1.0.10.in-addr.arpa ; origine du fichier (base ajoutée à toutes les entrées)

$TTL 86400 ; Valeur de la durée de validité des informations de la zone (ici 1 jour)

; début de la description de la zone (entrée SOA)
; nom_zone (@=valeur de $ORIGIN) IN SOA nom-serveur-fqdn. email-responsable.
@     IN     SOA    dns1.example.com.      hostmaster.example.com. (
                    2001062501 ; n° de série (doit être incrémenté après  chaque MAJ du fichier)
                    21600      ; Délai de MAJ esclave (ici 6 heures)
                    3600       ; Délai entre chaque essai de synchro du serveur esclave (ici 1h)
                    604800     ; Délai à partir duquel un serveur esclave arrête de répondre
                    86400 )    ; doublon avec $TTL

          IN     NS     dns1.example.com.
          IN     NS     dns2.example.com.

5         IN     PTR    server1.example.com. ; 10.0.1.5 ➔ server1
7         IN     PTR    server2.example.com. ; …
2         IN     PTR    dns1.example.com.
3         IN     PTR    dns2.example.com.
```

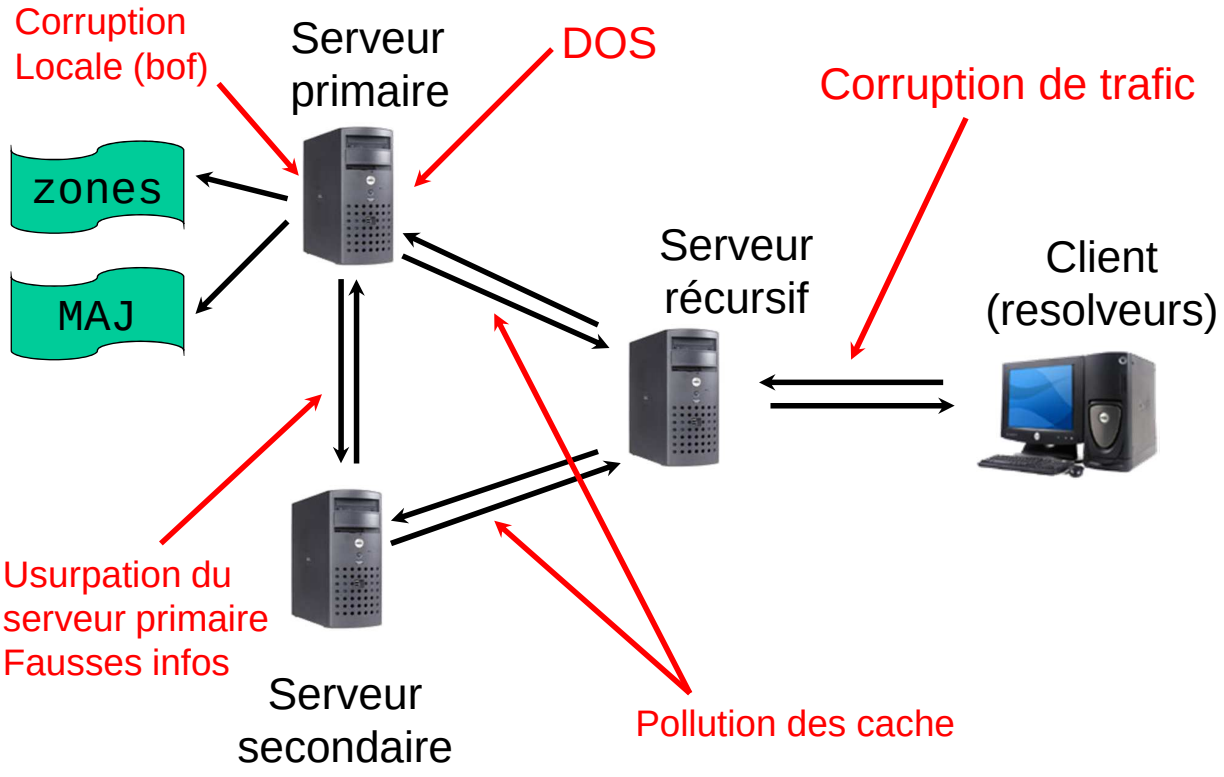
- Extension dns. ➔ annuaire de services !!!!!

Attaques sur DNS cache poisoning par spoofing

- DNS est un service de désignation !
- Le serveur DNS supporte tout les autres services.
- Une attaque du DNS permet de détourner une machine de sa cible !
  - Mise en place de faux sites WEB
  - Détournement d'informations
  - Attaques « Man In the Middle »
- Le DNS est vulnérable (RFC 3833)
  - Il ne repose quasiment que sur IP et UDP pour l'authentification de l'hôte pair

DNS

Attaques sur DNS cache poisoning



DNS

Attaques sur DNS cache poisoning par spoofing

- Le dialogue entre les serveurs DNS se passent de port 53 à port 53
- Le serveur traitent de nombreuses requêtes en UDP pour des raisons de performances
  - Il faut les distinguer
  - Pour cela, on utilise un identifiant de requête ID sur 16bits

ID	Options	Question	Réponse	divers

- Il est possible de s'insérer entre un client et son serveur DNS local
  - On sniffe la requête
  - On génère une fausse réponse avec l'ID sniffée
  - On l'envoie au client avant le serveur
  - La réponse du serveur sera ignorée

DNS

Attaques sur DNS cache poisoning par spoofing

- Si, on ne peut pas sniffer, l'attaquant peut essayer de prédire l'ID pour engendrer une attaque
- Sous Windows 95 ➔ L'ID est le nombre de req. DNS en cours !
- Bind ancienne version ➔ Nombre aléatoire puis incrémental
  - Méthode 1
    - ✔Il suffit d'une requête sur un DNS « sniffable » pour obtenir le point de départ !
  - Méthode 2
    - ✔L'attaquant demande une IP inexistante (ex: inconnu.domaine.com)
    - ✔Le DNS cible fait une requête « ns1.domain.com »
    - ✔L'attaquant génère une dizaine de réponses spoofées venant (soi-disant) de « ns1.domaine.com »
    - ✔ID allant de 200 à 210
    - ✔Si on obtient une réponse c'est qu'on a deviné l'ID
- Et si c'est aléatoire ?

DNS

Attaques sur DNS : birthday attack

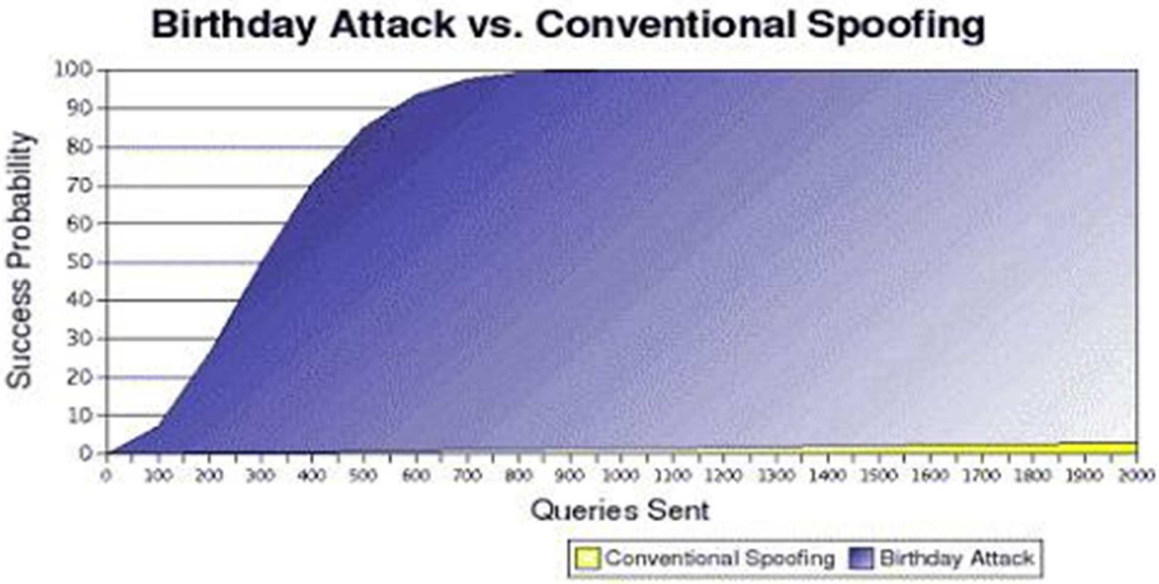
- Il est difficile de deviner l'ID d'une requête ($1/2^{16}$)
  - 1 requête et X réponses spoofées en temps très limité ➔ $p = X / 2^{16}$
- Birthday paradox attack !!!
  - Attaque basé sur un paradoxe apparent :

« sur une classe de 23 élèves ou plus, la probabilité que 2 élèves soient nés le même jour est supérieure à ½ »

- Technique appliquée au DNS
  1. Envoi de N requêtes à un serveur cache portant sur la même demande (www.exemple.com) associés à N IDs différents
  2. Transfert des N requêtes vers le serveur autoritaire du domaine exemple.com
  3. DoS sur le serveur autoritaire pour le ralentir
  4. Envoi de N réponses forgées associées à N IDs différents par l'attaquant
- Si N messages (~300) , t=le nombre de possibilités (2^{16})
  - la probabilité de succès de l'attaque $1-(1-1/t)^{N(N-1)/2}$
  - « p=.4956 » soit ~1/2

DNS

Attaques sur DNS : protection DNSsec



DNS

Attaques sur DNS : solutions

- Améliorer l'aléatoire de l'ID (espace des nombres)
- Split-Split DNS
  - FIREWALL : Interdire les IPs de votre domaine comme source sur votre point d'accès internet (paquets provenant de l'extérieur!)
  - Un serveur responsable du domaine
    - ✔déclaré et accessible de l'extérieur
    - ✔N'autorisé aucune requête récursive (hors domaine)
  - Un serveur cache DNS privé
    - ✔Autoriser requêtes récursives sur votre domaine seulement
- Déploiement de DNSSec

DNS

Attaques sur DNS : solution DNSsec et TSIG

- Sécurité des données et des transactions (MAJ)
- Architecture de distribution des clefs
  - Clefs utilisées par DNSsec
  - Clefs stockées dans le DNS sécurisé utilisées pour d'autres applications (IPsec, SSH)
- Sécurité des transactions (TSIG, RFC 2845)
  - Le transfert de zones
  - Les MAJ dynamiques (DNS Dynamic Updates)
  - Le canal entre serveur récursif et client
  - Authenticité forte, intégrité, protection rejeu
  - Pas de confidentialité

DNS

Attaques sur DNS : solution DNSsec et TSIG

- Sécurité des données (DNSSec, RFC 4033 à 4035)
  - DNSSec assure une chaîne de confiance
  - Chaque serveur a une clef
  - Chaque serveur peut identifier de manière forte les serveurs des sous-domaines de confiance
  - Inclus un protocole de MAJ des clefs
  - Ajoute deux types d'entrées
    - ✔SIG ➔ pour les signatures et KEY ➔pour les clefs privées

Attaques FTP : FTP servers bounce

- Le protocole FTP sépare le canal de contrôle (21) et le canal de téléchargement.
  - Il est possible « d'imposer » au FTP une adresse spécifique
  - PORT aa,bb,cc,dd,pp,qq ➔ ip(aa.bb.cc.dd), port (pp,qq)
- Cette option permet de contourner les limitations de téléchargement sur les IP
  - Fichiers protégés par la loi sur l'exportation US
- Mais il permet plus !

DNS

Section : « Administration réseau »

La base du réseau

Routage IP et couche liaison

Couche Transport : TCP/UDP

Configuration réseau

Outils réseau

Parefeu – NAT - SSL/TLS

DHCP

DNS

IDS et Analyse

DNS

« fwlogwatch » : Analyse de log firewall

- N'analyse que les paquets tracés
  - Nécessiter de bien définir les paquets à tracer
  - Les paquets jetés (drop/reject) sont à tracer
  - Certains paquets acceptés doivent être tracés
    - ✔Ouverture de connexion (TCP SYN)
  - Ne capte pas toute l'activer réseau de la machine !
- « fwlogwatch » :
  - Il génère des rapports
  - Il peut surveiller l'évolution des logs
    - ✔Générer des alertes via des scripts (EMAIL !)

fwlogwatch : Exemple

`fwlogwatch HTML`

- C'est joli les premières fois
  - Mais on se lasse vite
- Une fois en place
  - il faut prendre le temps chaque jour
  - Il faut être persévérant
- Si on ne le regarde plus, autant arrêter la génération des comptes rendus

Sonde IDS

- IDS = « Intrusion Detection System »
- Utilité des sondes IDS ?
  - Il est impossible de se protéger contre toutes les attaques
  - Détecter les attaques non bloquées
  - Empêcher l'espionnage de son réseau en le détectant précocement
  - Collecter le maximum d'information sur un attaque et sur les attaque (« know your ennemy »)
- Pour que la sonde soit efficace
  - Doit percevoir son environnement
  - Doit limiter ses interactions avec son environnement
    - ✔Eviter les détections de la sonde

Sonde IDS

- Type de sonde IDS
  - Sonde réseau ➔ analyse des trames
  - Sonde de machine ➔ analyse les événements machines
  - Sonde applicative ➔ analyse un service particulier (peut être une combinaison de plusieurs machines)
- La détection se fait par
  - Des signatures (comportementales)
  - La détection d'anomalies
- Certains IDS ont des capacités préprogrammées de réactions

Sondes IDS de type réseau

- NIDS = « Network based IDS »
- Type de sonde très employée
- Caractéristiques :
  - Elle écoute sur des points stratégiques du réseau
    - ✔Doit recevoir tout le trafic du réseau (dorsale)
    - ✔Penser à la brancher sur le routeur d'entrée et annuler l'isolation (ex: VLAN)
  - On peut segmenter les écoutes (plusieurs sondes)
  - Une sonde peut et doit être fortement sécurisée
    - ✔Contrôle d'accès fortement limité
    - ✔Hardened kernels
  - Une bonne configuration la rend difficilement détectable

Sondes IDS de type réseau

- Avantages
  - Peu de sondes bien placées peuvent surveiller un large réseau
  - Le déploiement de sonde a peu d'impact sur le réseau existant
    - ✔Peu d'effort de configuration pour déployer une sonde
  - Les sondes sont camouflables et bien protégées
- Inconvénients
  - Surcharge possible de la sonde (réseau, CPU)
    - ✔Certaines sondes utilisent un matériel spécifique
  - Certains commutateur bas de gammes n'offre pas de ports de surveillance (pas de copie du trafic)
  - Pour l'instant pas d'analyse de données cryptées
    - ✔Il faudrait connaître toutes les clefs et les protocoles de son réseau
  - Difficulté à savoir si une attaque à réussi ou non

IDS et analyse

Sondes IDS de type Machine

- Sonde IDS de type machine ➔ HIDS (Host based IDS)
- Caractéristiques
  - Analyse l'activité système de la machine sur laquelle elle est déployée
    - ✔« *System Integrity Verifier* » ➔ Vérification des modifications apportées sur les fichiers du système
    - ✔« *Log file monitor* » ➔ Vérification des traces systèmes
  - Permet de déterminer les activités suspectes de certains processus
  - Certains sondes peuvent transmettre les informations à un concentrateur
  - Certains sondes peuvent générer des messages réseaux (SNMP) ou des emails

IDS et analyse

Sonde IDS de type Machine

- Avantages
  - Détecte des attaques non détectables sur le réseau
  - Peut être utilisé dans des environnements « cryptés »
  - Peut être utilisé sur des réseaux commutés
- Inconvénients
  - Les bases doivent être mise à jour sur les machines
  - Les sondes peuvent être corrompues/désactivées par l'attaquant
  - Pas de vision globale (scans réseaux) car elle est déployée sur une machine « cliente »
  - La sonde machine consomme du CPU et de la mémoire sur la machine « cliente »
  - Difficulté à détecter les dénis de services

IDS et analyse

Sondes IDS de type applicative

- Très proches des sondes machines
- Souvent confondues avec les sondes machines
- Caractéristiques
  - Encapsule une application
  - Surveille une application et ses évènements
  - Spécialiser pour chaque type d'application
- Avantages
  - Très fine granularité pour analyser des comportements anormaux
  - Analyse les données après le décryptage
- Inconvénients
  - Sensibilité extrême aux attaques
  - Facilement corruptible

IDS et analyse

Sondes IDS : analyse par signatures

- Déclenchement sur la détection d'une suite prédéfinie d'évènements
- Méthode peu coûteuse et efficace
- Avantages
  - Peu de fausses alarmes
- Inconvénients
  - Il faut mettre à jour la base de signature
  - Ne détecte pas toujours toutes les variantes des attaques

IDS et analyse

Sondes IDS : analyse par détection d'anomalies

- Identification de comportements anormaux
- Un bon complément à l'analyse par signature
- Nécessité de définir une (la?) normalité
- Fonction d'apprentissage des comportements normaux
  - Analyse statistique des comportements
- Détection de déviance vis-à-vis de la normalité
- Avantages
  - Détection d'attaques non encore identifiées
- Inconvénients
  - Produits de nombreuses fausses alarmes
  - Nécessite un apprentissage couteux et long

IDS et analyse

Sondes IDS : réponses automatiques

- L'administrateur n'est pas toujours présent
- Les réponses passives automatiques peuvent être :
  - Envoie de notifications et déclenchement d'alarmes ➔ NECESSAIRE !
    - ✔ Email, sms, ... aux responsables
  - Envoie de trames SNMP
  - Utilisation de plug-in
  - Archivage automatique sur un support sûr
- Les réponses actives peuvent être
  - Changer l'environnement interne ➔ PEUT ETRE DANGEREUX ET ETRE EXPLOITE !
    - ✔ Injection de trames RST pour couper l'attaque
    - ✔ Reconfiguration des routeurs et firewall pour bloquer @IP
    - ✔ Reconfiguration des routeurs et firewall pour bloquer les protocoles
    - ✔ Bloquer totalement le réseau dans les cas extrêmes
  - Collecte automatique d'informations sur la source ➔ réponse la plus efficace
    - ✔ Whois, nmap, traceroute
    - ✔ Changer le niveau de logging de l'IDS, activer d'autres type de sonde
  - Attaquer l'attaquant ! ➔ réponse la plus dangereuse (légalement et techniquement)
    - ✔ Interdit par la loi et peut attaquer quelqu'un d'innocent

IDS et analyse

Sondes IDS : limitations ?

- Etre conscient des limitations des IDS
- Difficultés d'estimer les ressources nécessaires
  - Ressources CPU/Réseau
  - Ressources humaines pour traiter les alarmes et les MAJ des IDS
- Génération de faux positifs coûteux pour les administrateurs systèmes
- Les IDS mêmes considérés comme temps-réels mettent parfois plusieurs minutes à réagir ➔ dû à la charge de la machine
  - Temps extrêmement long en informatique
- Latence entre la mise à jour des bases et le déploiement
  - Attaque non détectée possible entre la publication et la MAJ de l'IDS
- Les réponses automatiques
  - Sont souvent ineffectives contre les hackers expérimentées
  - Peuvent gêner le trafic légal
- Les IDS ne sont pas forcement protégés contre les attaques
- Les IDS n'ont pas toujours de GUI et d'outils d'analyse efficaces

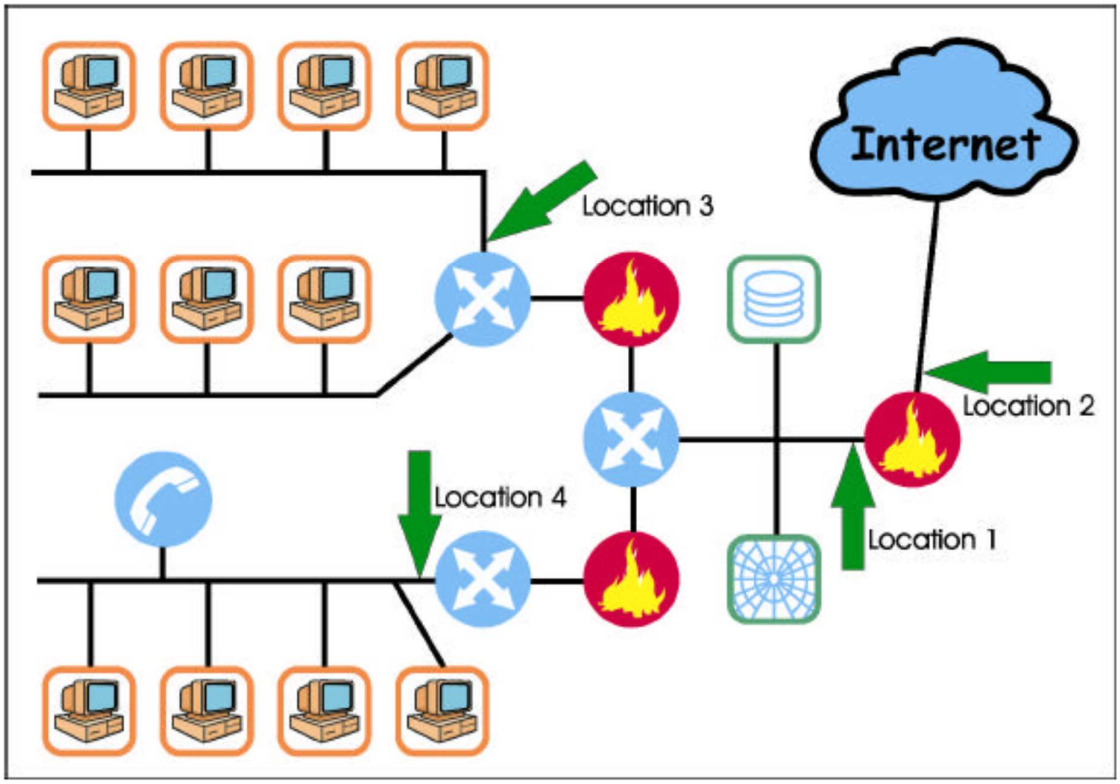
IDS et analyse

Sondes IDS : Localisation

- Localisations de déploiement
  - Derrière chaque firewall externe
  - Dans le réseau DMZ
  - Devant le firewall frontale (point d'entrée)
  - Sur les dorsales internes des sous-réseaux
- Ce sont des conseils !
  - Vous devez adapter votre stratégie IDS à
    - ✔Votre réseau
    - ✔Aux ressources financières
  - C'est un domaine encore en phase de R&D

IDS et analyse

Sondes IDS : Localisation



IDS et analyse

NIDS payants

- Network ICE, BlackICE, Win
  - Auto-update, TCPIP/ARP, SMNP
- Network Associate, Cybercop Monitor, Win
  - Product update, TCPIP, pas SNMP
- Cisco, Netranger, Solaris
  - MAJ par CD, TCPIP, SNMP
- ISS, RealSecure, Win/Unix/Linux
  - MAJ par HTTPs, TCPIP, SNMP
- AXENT, Omniguard, Win/AIX/Unix/HP/Solaris
  - MAJ par HTTPs, pas SNMP
- . . .

NIDS gratuits

- Les applications libres
  - Snort, Prelude, BroIDS, hogwash ➔ Sondes de type réseau
  - Nessus, ettercapNG ➔ dans le domaine de l'audit de vulnérabilités
  - IDSwakup ➔ permet de générer du trafic réseau anormal
  - Nmap, dsniff, kismet ➔ scan réseau
  - Argus ➔ Nework logger
- Elles font partie des références incontestables dans leur domaine.
  - performances parfois supérieures à celle d'applications commerciales souvent vendues très cher.
- Problème :
  - La société éditrice de Snort vient d'être rachetée
  - L'auteur de nessus ne veut plus publier ses sources
- Ils sont quand même à déployer car incontournables !
  - En attendant mieux ou leurs récupérations open source
- Il faut déployer PLUSIEURS IDS SIMULTANEEMENT !!!!!

IDS snort : configuration

- Installation
  - Edition du fichier /etc/snort/snort.conf
  - Modifier la variable HOME_NET pour définir le périmètre de confiance (ex: votre réseau)
  - L'extérieur est définit comme ! HOME_NET
  - Activation possible du log dans une DB ou un fichier binaire (bainard)
    - ✔ Variable « output »
    - ✔Permet des débits plus importants que le texte pur
  - Activer tous les plugins (*preprocessors*) et les règles (*include*) qui vous intéressent
    - ✔Préprocesseurs: sfPortscan, stream4, httpinspect, rpcdecode
    - ✔En cas de réseau exposé, activer les règles BleedindEdge
    - ✔Les règles de bases sont à MAJ sur Arachnids
    - ✔Fixer les options
- /etc/init.d/snort start

IDS et analyse

IDS snort : outils

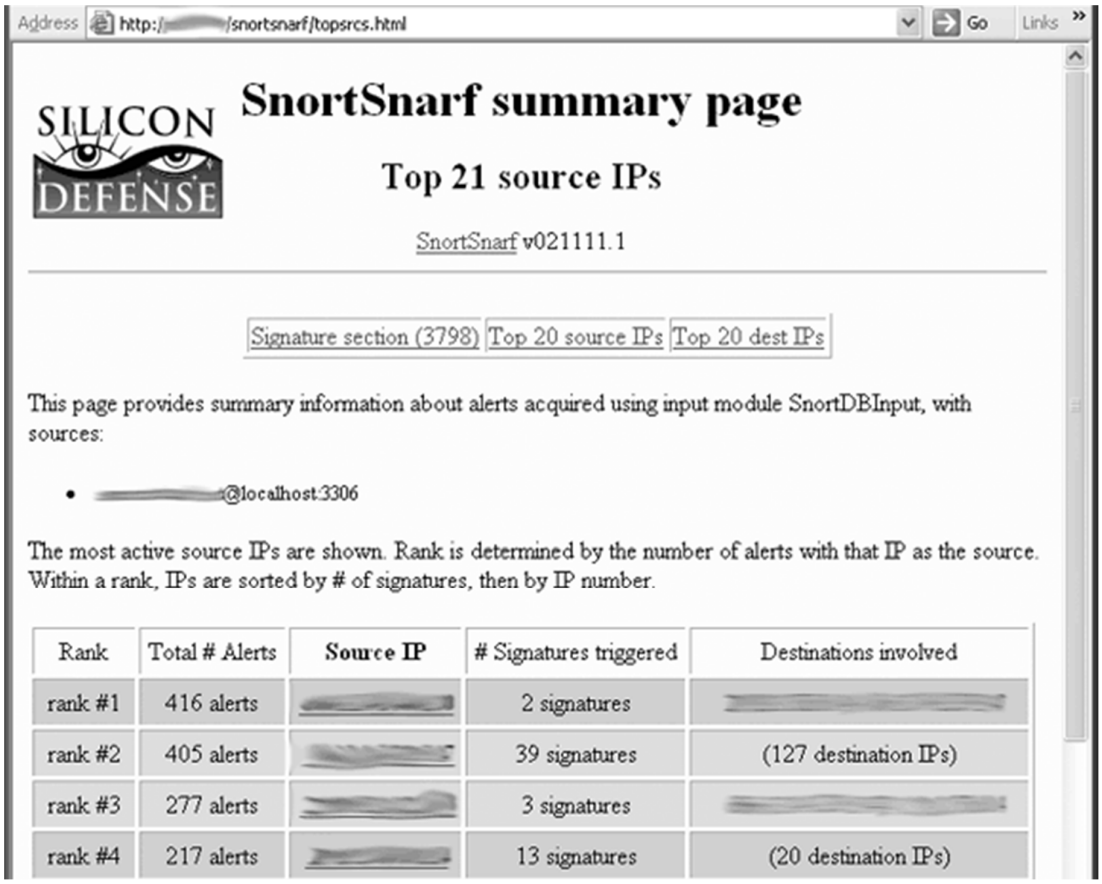
- Snort ne fait que tracer les attaques
  - Pas d'email
  - Pas de modification des règles de firewall
    - ✔Pour iptables ➔ module snortsam ou snort-inline
- Pour emailer les alertes snorts
  - Swatch, IDSCenter, logsurfer
- Analyse et gestion des logs pour snort
  - Snortsnarf produit un rapport HTML à partir des logs snort
  - ACID produit un rapport HTML à partir d'une BD snort
  - Cerebus pour analyser les logs
  - Autres: 5n0r7, SnortReport, SnortBot, SnortPHP, snort_stat.pl (livré avec snort)

IDS et analyse

IDS snort : outils

- GUI pour snort
  - HenWen (MacOSX), IDSCenter (Win), SnortCenter (Linux/Win)
- Maintenance des logs snort:
  - Guardian, logsnorter, snortlog
- Outils
  - Getcontact, Hogwas Signature
- Configuration snort
  - IDS Policy Manager, Snort Webmin Module
- Mise à jour des règles snort
  - oinkmaster
- Pour obtenir les modules Snort : http://www.snort.org/downloads.html

IDS snort : Exemple snortsnarf



IDS snort : Exemple snortsnarf

Snortsnarf HTML

- Comme fwlogwatch
  - c'est joli les premières fois ➔ Mais on se lasse vite
  - Une fois en place ➔ il faut prendre le temps chaque jour
- Si on ne le regarde plus, autant arrêter la machine
- Exemples :
  - Summary : connexion BO
  - Summary : trafic hors norme (UDP 53 vers 139 !!)
  - Top 20 source IP (211.137.96.156) : Host zombie typique
    - ✔1 type d'attaque, répéter régulièrement
  - Top 20 source IP (216.136.86.44) : Scan de proxy web/socks
  - Top 20 destination IP ➔ machines les plus ciblées

IDS et analyse

IDS snort : Exemple ACID

Analysis Console for Intrusion Databases

Added 20 alert(s) to the Alert cache

Queried on : Wed July 31, 2002 15:35:26
Database: snort@localhost (**schema version:** 105)
Time window: [2002-07-21 20:20:31] - [2002-07-31 15:35:22]

Sensors: 1
Unique Alerts: 4 (2 **categories**)
Total Number of Alerts: 83

- Source IP addresses: 3
- Dest. IP addresses: 4
- Unique IP links 4
- Source Ports: 4
  - TCP (3) UDP (1)
- Dest. Ports: 2
  - TCP (1) UDP (1)

Traffic Profile by Protocol
TCP (4%)
UDP (48%)
ICMP (48%)
Portscan Traffic (0%)

- Search
- Graph Alert data (EXPERIMENTAL)

- **Snapshot**
  - Most recent Alerts: any protocol, TCP, UDP, ICMP
  - Today's alerts unique, listing; IP src / dst
  - Last 24 Hours: alerts unique, listing; IP src / dst
  - Last 72 Hours: alerts unique, listing; IP src / dst
  - Most recent 15 Unique Alerts
  - Last Source Ports: any , TCP , UDP
  - Last Destination Ports: any , TCP , UDP
- Most frequent 5 Alerts
- Most Frequent Source Ports: any , TCP , UDP
- Most Frequent Destination Ports: any , TCP , UDP
- Most frequent 15 addresses: source, destination
- Graph alert detection time

IDS et analyse

IDS snort : Exemple ACID

ACID Query Results: 15 Last ICMP Alerts

Home | Search | AG Maintenance

[Back]

Added 23 alert(s) to the Alert cache

Meta Criteria Last 15 ICMP Alerts

Summary Statistics
- Sensors
- Unique Alerts (classifications)
- Unique addresses: source | destination
- Unique IP links
- Source Port: TCP | UDP
- Destination Port: TCP | UDP
- Time profile of alerts

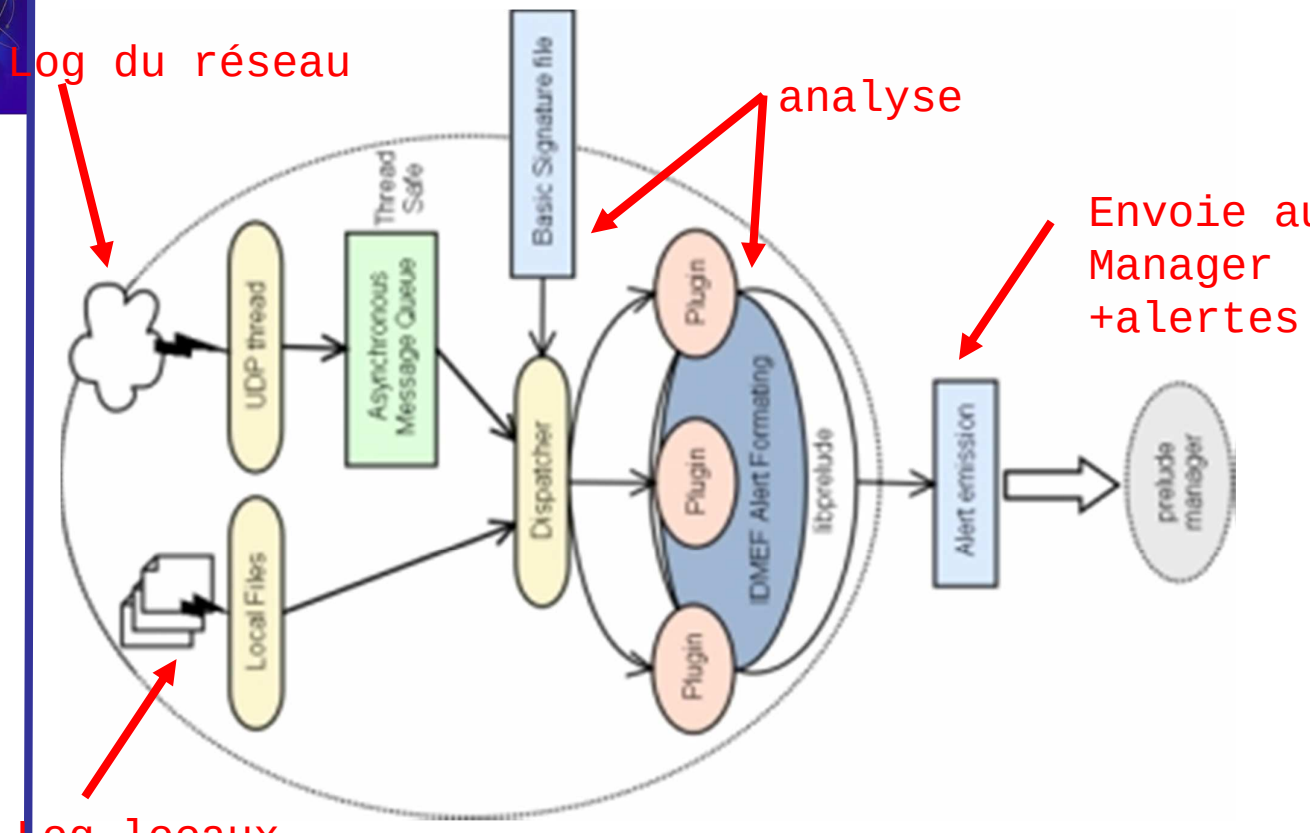
Displaying 15 Last ICMP Alerts

ID	< Signature >	< Timestamp >	< Source Address >	< Dest. Address >	< Layer 4 Proto >
#0-(1-43)	Dont Fragment bit set	2002-07-31 15:33:34	192.168.1.2	192.168.1.100	ICMP
#1-(1-42)	ICMP Packet with TTL=100	2002-07-31 15:33:34	192.168.1.100	192.168.1.2	ICMP
#2-(1-41)	Dont Fragment bit set	2002-07-31 15:33:33	192.168.1.2	192.168.1.100	ICMP
#3-(1-40)	ICMP Packet with TTL=100	2002-07-31 15:33:33	192.168.1.100	192.168.1.2	ICMP
#4-(1-39)	Dont Fragment bit set	2002-07-31 15:33:32	192.168.1.2	192.168.1.100	ICMP
#5-(1-38)	ICMP Packet with TTL=100	2002-07-31 15:33:32	192.168.1.100	192.168.1.2	ICMP
#6-(1-37)	Dont Fragment bit set	2002-07-31 15:33:31	192.168.1.2	192.168.1.100	ICMP
#7-(1-36)	ICMP Packet with TTL=100	2002-07-31 15:33:31	192.168.1.100	192.168.1.2	ICMP
#8-(1-35)	Dont Fragment bit set	2002-07-23 18:17:40	192.168.1.2	192.168.1.100	ICMP

IDS: Prelude

- « Prelude » est similaire à snort
- Il est composé de 3 type de serveurs
  - Un serveur manager ➔centralise les informations collectées par les sondes
    - ✔« prelude-manager », fichiers « /etc/prelude-manager/* »
  - Un serveur sonde NIDS ➔ collecte des informations réseaux en un point du réseau
    - ✔« prelude-nids », fichiers « /etc/prelude-nids/* »
  - Un serveur sonde LML ➔ collecte des informations sur des machines (les logs)
    - ✔« prelude-lml », fichiers « /etc/prelude-lml/* »
  - Un serveur de contre-mesure ➔ gére les actions de contre-mesures en fonction de données collectés par les managers
    - ✔« prelude-cm-agent »
- Les communications entre les sondes et le manager est sécurisé

IDS: serveur « prelude-lml »



IDS et analyse

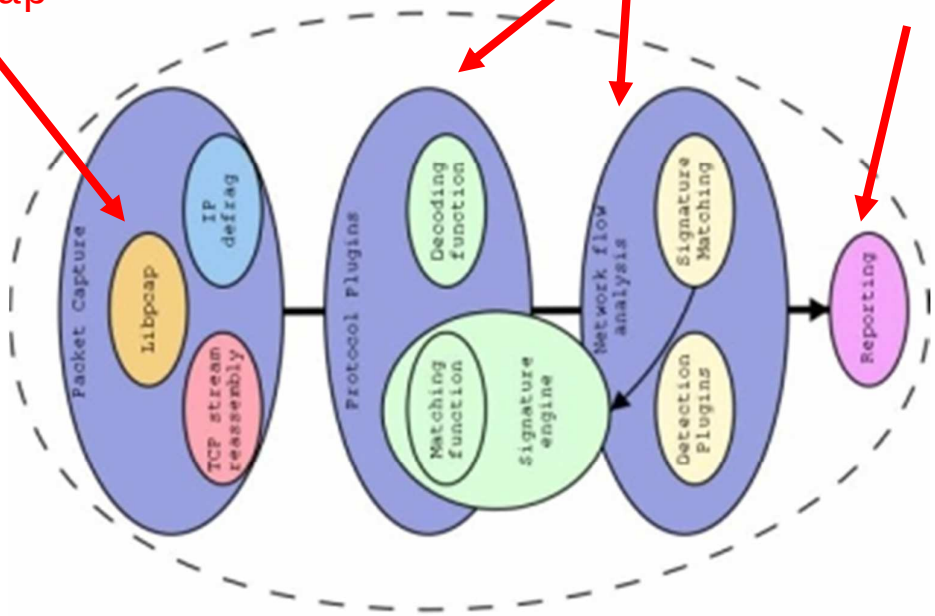
IDS: serveur « prelude-nids »

Capture par libpcap

analyse

Envoie au Manager +alertes

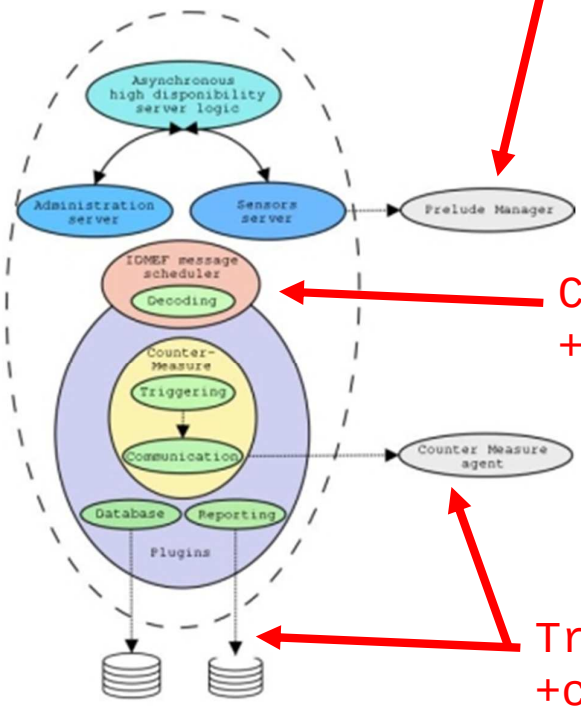
Module réseau



IDS et analyse

IDS: serveur « prelude-manager »

Envoie au Manager +alertes



Centralisation +décision

Tracabilité +contre-mesures

IDS Prelude : ajout d'une sonde

- Pour l'installation, les paquetages nécessaires sont
  - « prelude-manager », « prelude-tools »
  - « prelude-nids », « prelude-lml »
  - Eventuellement une BD (mysql) et « prelude-cm-agent »
- Prelude n'accepte pas les données de sources inconnues
  ➔ il faut gérer un échange de clefs
- Pour chaque serveur, pensez au fichier /etc/prelude-xxx/prelude-xxx.conf
- Pour ajouter une sonde, il faut du côté « manager »

```
# manager-adduser
Generated one-shot password is "sa17bh20".
This password will be requested by "sensor-adduser" in order to connect.
Please remove the first and last quote from this password before using it.
- Waiting for install request from Prelude sensors...
```


IDS Prelude : ajout d'une sonde

- Du coté sonde : « -u » uid, « -s » nom sonde, « -m » ip manager

```
# sensor-adduser -s sensor-lml -m 127.0.0.1 -u 0

Now please start "manager-adduser" on the Manager host
where you wish to add the new user.
Please remember that you should call "sensor-adduser"
for each configured Manager entry.
Press enter when done.
Please use the one-shot password provided by the
"manager-adduser" program.

Enter registration one shot password : sa17bh20
Please confirm one shot password : sa17bh20

connecting to Manager host (127.0.0.1:5553)...
Succeeded.

Username to use to authenticate : sensor-lml
Please enter a password for this user : password
Please re-enter the password (comfirm) : password
Register user “sensor-lml" ? [y/n] : y
Plaintext account creation succeed with Prelude Manager.
```

IDS Prelude : résultats collectés

- Lancer le manager (« prelude-manager ») et les sondes (NIDS et LML)
- Le fichier de log est « /var/log/prelude-manager/prelude.log »
- Il existe, comme pour snort, des générateurs de rapport
- Au démarrage de prelude-nids

```
************************************************************
* Heartbeat: ident=1
* Analyzer ID: 1041315032505060971
* Analyzer model: Prelude NIDS
* Analyzer version: 0.8.6
* Analyzer class: NIDS
* Analyzer manufacturer: The Prelude Team http://www.prelude-ids.org
* Analyzer OS type: Linux
* Analyzer OS version: 2.6.12-12mdk-i686-up-4GB
* Node[unknown]:
* Process: pid=26218 name=prelude-nids
* Creation time: 0xc76582cb.0x94c4300 (2006-01-04 00:10:03.581+0100)
*
************************************************************
```

- Même type de message au démarrage de prelude-lml

IDS Prelude : résultats collectés (NIDS)

```
********************************************************************************
* Alert: ident=3923
* Classification type: bugtraqid
* Classification: BAD-TRAFFIC IP Proto 103 (PIM)
* Classification URL: http://www.securityfocus.com/bid/8211
* Creation time: 0xc7658fe6.0xc659400 (2006-01-04 01:05:58.774+0100)
* Detection time: 0xc7658fe6.0xc656a00 (2006-01-04 01:05:58.774+0100)
* Process: pid=26372 name=prelude-nids path=/usr/bin
* Impact severity: medium
* Impact completion: NULL
* Impact type: other
* Impact description: Detection of a non-standard protocol or event
*** Source information ************************************************************
* Source spoofed: unknown
* Node[unknown]:
* Addr[ipv4-addr]: 132.227.64.15
*** Target information ************************************************************
* Target decoy: unknown
* Node[unknown]:
* Addr[ipv4-addr]: 224.0.0.13
*** Additional data within the alert  ***************************************
* Ethernet header: 0:10:d:3d:c4:0 -> 1:0:5e:0:0:d [ether_type=ip (2048)]
* Ip header: 132.227.64.15 -> 224.0.0.13 [hl=20,version=4,tos=192,len=38,id=26960,ttl=1,prot=103]
* Payload header: size=18 bytes
* Payload Hexadecimal Dump: 20 00 c9 b0  00 01 00 02  00 69 00 14 00 04 00 00 15 cb
* Detection Plugin Name: SnortRules
* Detection Plugin Author: The Prelude Team
* Detection Plugin Contact: prelude-devel@prelude-ids.org
* Detection Plugin Description: Snort signature parser.
* Snort rule ID: 2189
* Snort rule revision: 1
********************************************************************************
```

Protocole de routage multicast CISCO

IDS et analyse

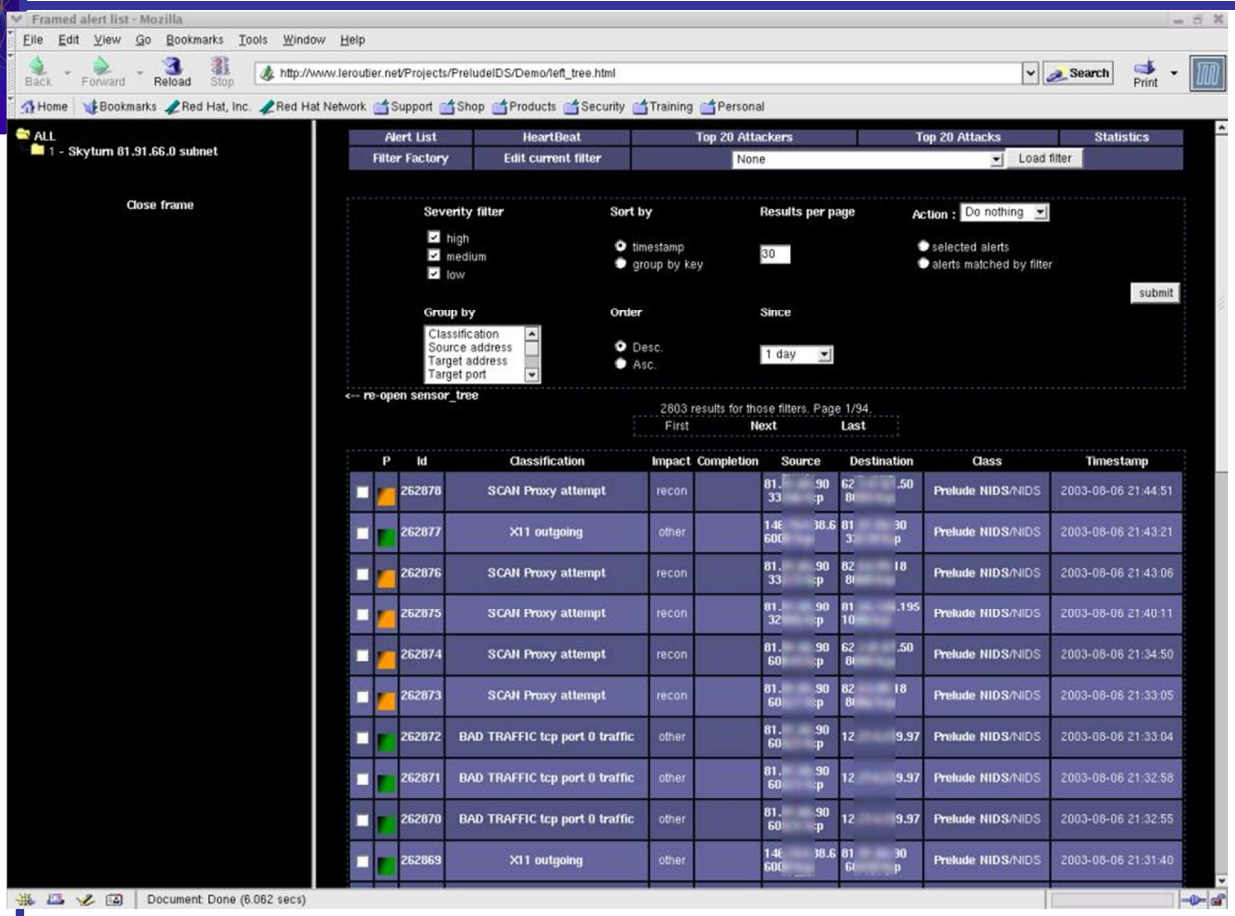
IDS Prelude : résultats collectés (LML)

```
********************************************************************************
* Alert: ident=4042
* Classification type: unknown
* Classification: SSH Remote root logging failed
* Classification URL: unknown
* Creation time: 0xc7659e1c.0x8b7a000 (2006-01-04 02:06:36.544+0100)
* Detection time: 0xc7659e1c.0x0000000 (2006-01-04 02:06:36.000+0100)
* Analyzer ID: 630008679108729663
* Analyzer model: Prelude LML
* Analyzer version: 0.8.6
* Analyzer class: HIDS
* Impact severity: medium
* Impact completion: failed
* Impact type: admin
* Impact description: Someone tried to login as root from 132.227.64.30:34689 using the password method
*** Source information ********************************************************
* Source spoofed: unknown
* Node[unknown]:
* Addr[ipv4-addr]: 132.227.64.30
* Service: port=34689 protocol=tcp
*** Target information ********************************************************
* Target decoy: unknown
* Service: port=22 protocol=tcp
* Target decoy: unknown
* Node[unknown]: name:eos.lip6.fr
* Addr[ipv4-addr]: 132.227.64.45
* Process: pid=0 name=sshd
*** Additional data within the alert  *****************************************
* Log received from: /var/log/messages
* Original Log: Jan  4 02:06:36 eos sshd[27082]: Failed password for root from 132.227.64.30 port 34689 ssh2
********************************************************************************
```

Tentative de login Ssh en « root »

IDS et analyse

IDS Prelude : rapport HTML (piwi)



Détection de vulnérabilité

- « nessus », « ettercap » permettent de faire de la détection de vulnérabilité (des attaques)
- « nmap » est un bon complément
- « IDSwakeup » génère du trafic anormal pour déclencher une réaction des IDS
- A faire régulièrement ➔ Au moins une fois par mois