

---

# Mise en œuvre de Rembo Toolkit

- Pré requis serveurs
- Fonctionnement du service
- Configuration liée a PXE
- Tunnels TCP
- Installation/config
- Outils Rembo
- Les clients

---

# Pré requis serveurs

- Os possible : Windows, Linux, FreeBSD, MacOSX, Solaris
- CPU : minimum 512M
- Disque : minimum 20G  
a accès rapide : SCSI ...
- Carte réseau Giga (si possible 1 par sous-réseau)

---

# Pré requis serveurs

- Service DHCP
  - Si on garde un existant : pas de configuration spéciale mais enlever l'existant options 43, 60
  - Si on place le dhcp sur le serveur rembo créer l'option 60 (class identifier) = « PXEClient »
  - penser à la variable « Hostname »
  - Ordre de démarrage : dhcp puis rembo

---

# Pré requis serveurs

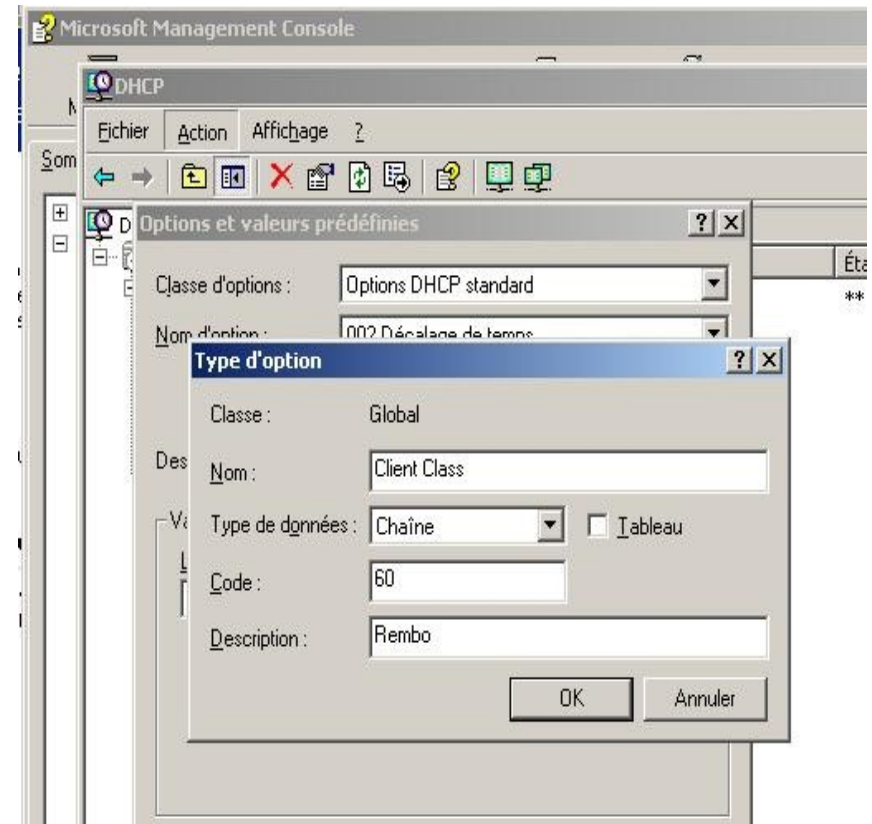
## Ex : Service DHCP Windows

- Installer : ajout/suppression de programmes des composants Windows
- Service de mise en réseau : Protocole DHCP
- Dans le menu gérer le serveur, menu gérer le DHCP

# Pré requis serveurs

## Ex : Service DHCP Windows 2003

- Créer l'option 60 :  
Serveur / définir les options prédéfinies->  
Ajouter
  - Nom: "Client Class"
  - Type: "Chaîne"
  - Code : 60
  - Valeur : "PXEClient"
- Ajouter option 60 avec la valeur "PXEClient" pour une étendue, une réservation



---

# Pré requis serveurs

## Ex : ISC DHCP - Fedora 4

- Récupérer la dernière version sur le site <http://www.isc.org/index.pl?/sw/dhcp/>
- `gunzip -c src/dhcp-3.0.3.tar.gz |tar -xf -`  
`cd /home/dhcp-3.0.3/`  
`./configure`  
`make`  
`make install`  
`touch /var/state/dhcp/dhcpd.leases`

# Pré requis serveurs

## Ex : ISC DHCP - Fedora 4

### ■ Configurer vos client /etc/dhcpd.conf

```
ddns-update-style ad-hoc;
default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0;
option routers 192.168.23.254; option
domain-name-servers 192.168.23.1;
option domain-name "mydomain.org";

subnet 192.168.23.0 netmask 255.255.255.0 {
    option vendor-class-identifier "PXEClient";

    host fc4client {hardware ethernet 00:0C:29:A9:47:7A; fixed-address
192.168.23.135; option Host-name "fc4client";}
    host winclient {hardware ethernet 00:0C:29:0C:04:CE; fixed-address
192.168.23.136; option Host-name "winclient";}

    range 192.168.23.140 192.168.23.150;
}
```

---

# Pré requis serveurs

## Ex : ISC DHCP - Fedora 4

- Créer un script de démarrage  
`/etc/init.d/dhcpd`
- Démarrer le service  
`/etc/init.d/dhcpd start`
- Penser au reboot du serveur, le service doit démarrer automatiquement avant rembo  
`chkconfig --list dhcpd`



# Pré requis serveurs

- Base de données : rembo a des commandes pour accéder et modifier des bases de données ODBC.
- Serveur Windows, une passerelle s'installe automatiquement en temps que service.
- Serveur Windows avec rembo.conf, il faut lancer le service manuellement : `dbgw.exe -d`
- Serveur Linux, la passerelle est une application java qui doit être installée et démarrée avant le service rembo
- Il faudra créer un tunnel via rembo.conf ou la console web

# Pré requis serveurs

## Ex : Base de données – Fedora 4

- --> JAVA

```
rpm -i jdk-1_5_0_06-linux-i586.rpm
```

```
JAVA_HOME=/usr/java/jdk1.5.0_06
```

```
export JAVA_HOME
```

```
PATH=$PATH:$JAVA_HOME/bin
```

- -->MYSQL

```
rpm -i mysql-4.1.11-2.i386.rpm
```

```
rpm -i mysql-server-4.1.11-2.i386.rpm
```

# Pré requis serveurs

## Ex : Base de données – Fedora4

- --> Passerelle

```
gunzip -c src/mysql-connector-java-3.1.12.tar.gz |tar -xf -  
cd /usr/local/rembo-4.0
```

```
ln -s /home/mysql-connector-java-3.1.12/mysql-  
connector-java-3.1.12-bin.jar mysql.jar
```

- --> Lancement a effectuer avant le démarrage de rembo

```
java -cp dbgw.jar:mysql.jar -Djdbc  
.drivers=com.mysql.jdbc.Driver com.rembo.dbgw.Dbgw  
-d
```

# Pré requis serveurs

## Ex : Base de données – Fedora 4

- --> Config MYSQL

```
mysql_install_db --user=mysql
```

```
cd /var/lib/mysql/mysql
```

```
chown mysql *
```

```
mysqld_safe --user=mysql &
```

```
mysql -u root
```

```
delete from mysql.user where user="";
```

```
update mysql.user set password=password('xyz');
```

```
flush privileges;
```

```
exit
```

# Pré requis serveurs

## Ex : Base de données – Fedora 4

- --> créer la base

```
create database myRemboDatabase;
```

```
use myRemboDatabase;
```

```
CREATE TABLE toto (name VARCHAR(20),surname  
VARCHAR(20));
```

Aide mysql <http://dev.mysql.com/doc/refman/4.1/en/index.html>

- --> test

```
telnet localhost 2020
```

```
use mysql://127.0.0.1/myRemboDatabase,root,xyz
```

---

# Pré requis serveurs

- L'authentification
  - ❑ Serveur Windows : local, autre Windows ou Radius
  - ❑ Serveur Unix : local, NIS, NIS+, pam ou radius
  - ❑ Rembo authentifiera tous les utilisateurs d'un groupe sans distinction
  - ❑ Il faut créer un tunnel :

# Pré requis serveurs

- Avec un rembo.conf

```
AuthLocalDomain mesutilisateurs {UserGroup "users"}
GROUP salleA {
    AuthDomain " mesutilisateurs "
    Host aa:bb:cc:dd:ee:ff
}
```

- Sans rembo.conf : il faudra définir un domaine d'authentification dans la console web  
<http://localhost:8080>  
server parameter>predifined channels > "New auth. Domain"

## Pré requis serveurs

### Ex : Authentification ldap – Fedora 4

- Créer un fichier rembo dans /etc/pam.d  
auth required /lib/security/pam\_ldap.so  
account required /lib/security/pam\_ldap.so
- Configurer /etc/ldap.conf  
host monldapserveur.univ-rennes1.fr  
base ou=people,dc=univ-rennes1,dc=fr  
pam\_filter &(atrib1=oui)(attrib2=non)  
ssl no  
pam\_password md5



---

# Fonctionnement du service

- L'installation est simple : la configuration pourra être modifiée par la suite
- Il faut le fichier d'installation, la clé rembo.key, un mot de passe pour l'administration du service

# Fonctionnement du service

## ■ L'arborescence installée :

### /home/rembo-x/rembo.conf

- Il définit les différents groupes avec les adresses mac des postes qui les compose, leur page de démarrage, le mode d'authentification et la machine qui peut prendre la main. Dans la version 4.0 tout ceci est dans la base interne de rembo

### /home/rembo-x/logs

- Contient les fichiers de logs par adresse mac et file.log, boot.log, nbp.log et tcp.log (visibles dans les consoles serveur)

### /home/rembo-x/files/shared

- Contient les fichiers partagés par les images (ne pas y toucher).

### /home/rembo-x/files/global

- Contient tous les fichiers accessibles et modifiables à partir d'un client (visibles dans les consoles serveur)

---

# Fonctionnement du service

## /home/rembo-2.0/files/global/hosts

- Un répertoire par machine avec son adresse mac

## /home/rembo-2.0/files/global/groups

- Un répertoire par groupe définis dans le rembo.conf

## /home/rembo-2.0/files/global/scripts

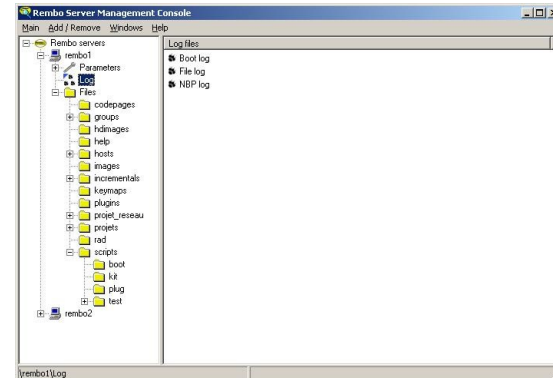
- Contient les scripts appelés par la startpage

# Fonctionnement du service

- Les consoles serveur

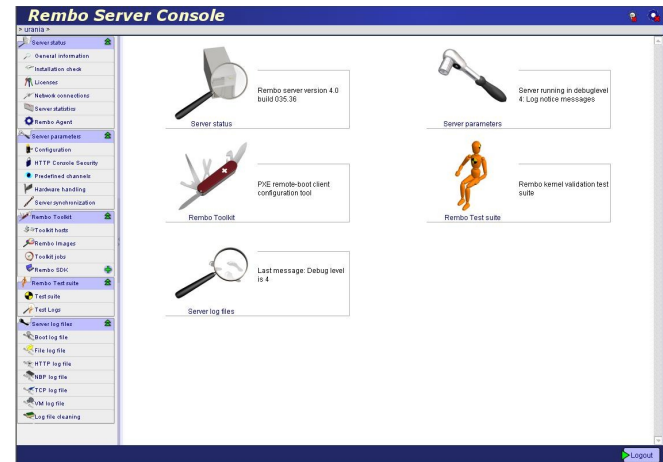
- Rembo 2.0

- Sconsole.exe



- Rembo 4.0

- <http://serveur:8080>



---

# Ex : Installation Windows

- Lancer l'exécutable et répondre aux questions

# Ex : Installation – Fedora 4

- ❑ `gunzip -c $SRC/Rembo-Toolkit-Install-4.0.034.tar.gz |tar -xf -`
- ❑ `cp rembo4.0.key /home/rembo-4.0/rembo.key`
- ❑ `chown -R root:root /home/rembo-4.0`
- ❑ `cd /home/rembo-4.0`
- ❑ `./setup`  
Répondre aux différentes questions, dans un premier temps les valeurs pas défaut peuvent suffire
- Lancer une première fois `./rembo -d -v 4`
- Ajouter un script de démarrage fournit par l'installation

# Configuration liée a PXE

- Le dhcp et les différents protocoles utilisés par rembo mtftp ... doivent passer les différents vlans
- Au niveau des différents commutateurs
  - Pour les interfaces spanning-tree portfast
- Au niveau du cœur de réseau
  - Ip helper-address ipmonserveur
- Acls
  - In
    - Permit udp host 0.0.0.0 host 255.255.255.255 eq 67
    - Permit igmp any any
    - Permit ip any
  - Out
    - Permit ip host reboserveur any
    - Permit igmp host reboserveur any

---

# Tunnels TCP

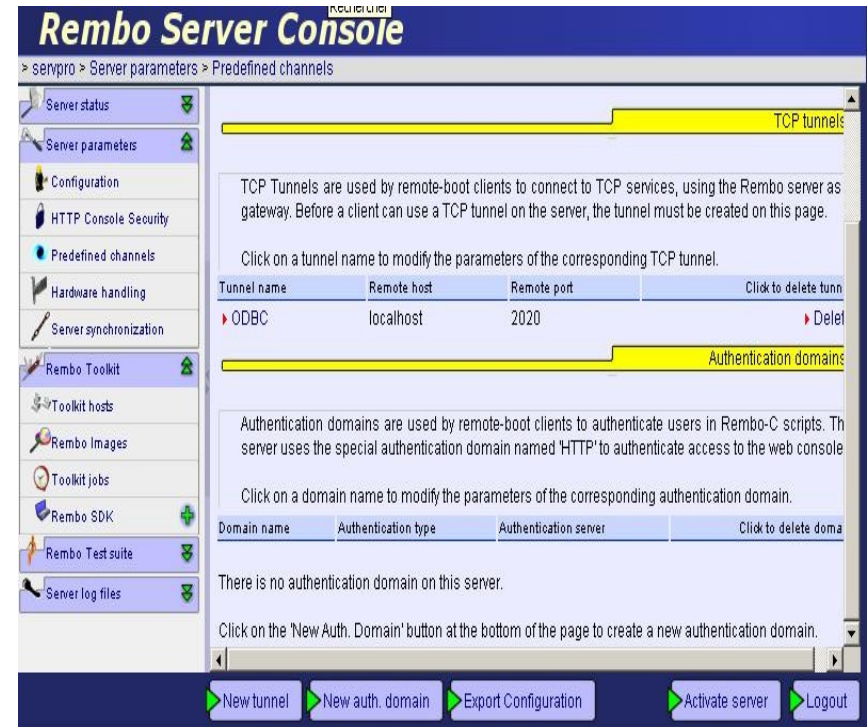
- Pour créer un tunnel, il faut que le service existe et fonctionne, il lui faut un nom un serveur et un port
- Un autre serveur rembo utilisera le port 4013
- Rembo.conf

```
TCPTunnel sendmail {  
    RemoteHost smtp.univ-rennes1.fr  
    RemotePort 25  
}
```



# Tunnels TCP

- Console web  
<http://localhost:8080>
  - server  
parameter>predifined  
channels > "new tunnel"
  - Nom: sendmail
  - Serveur: smtp.univ-  
rennes1.fr
  - Port: 25



The screenshot shows the 'Rembo Server Console' web interface. The breadcrumb navigation is 'servpro > Server parameters > Predefined channels'. The left sidebar contains various menu items: Server status, Server parameters, Configuration, HTTP Console Security, Predefined channels, Hardware handling, Server synchronization, Rembo Toolkit, Toolkit hosts, Rembo Images, Toolkit jobs, Rembo SDK, Rembo Test suite, and Server log files. The main content area is titled 'TCP tunnels' and contains the following text: 'TCP Tunnels are used by remote-boot clients to connect to TCP services, using the Rembo server as gateway. Before a client can use a TCP tunnel on the server, the tunnel must be created on this page. Click on a tunnel name to modify the parameters of the corresponding TCP tunnel.' Below this is a table with columns 'Tunnel name', 'Remote host', 'Remote port', and 'Click to delete tunnel'. The table contains one entry: 'ODBC', 'localhost', '2020', and a 'Delete' link. Below the table is the 'Authentication domains' section, which contains the text: 'Authentication domains are used by remote-boot clients to authenticate users in Rembo-C scripts. The server uses the special authentication domain named 'HTTP' to authenticate access to the web console. Click on a domain name to modify the parameters of the corresponding authentication domain. There is no authentication domain on this server. Click on the 'New Auth. Domain' button at the bottom of the page to create a new authentication domain.' At the bottom of the page, there are five buttons: 'New tunnel', 'New auth. domain', 'Export Configuration', 'Activate server', and 'Logout'.

# Outils Rembo

- Rembo 2.0
  - NetClient : accéder aux fichiers du serveur exporter des images en fichier.rad
  - Rconsole : prendre la main sur le client pendant l'exécution des scripts rembo
- Rembo 4.0
  - Packages : créer de nouvelles pages à la console web
  - Rembo Agent : service sur le client indépendant de l'OS permettant des actions en ligne à partir du serveur. Ex les tâches planifiées

---

# Les clients : pré requis

- Carte PXE > 2.0 PXE 0,99
- Interface graphique du bios VEGA mais VGA suffit
- Disques ATA et non-ATA si interface bios
- Système de fichiers et OS supporté : comme pour les serveurs

# Les clients : configuration

- Au niveau du Bios
- activer la carte réseau en PXE
- Dans la séquence de boot placer la carte réseau en premier
- Récupérer les informations réseau nécessaires pour le serveur @Mac ou ip
- Déclarer le client au niveau du serveur pour le placer dans un groupe pour qu'il utilise la page de démarrage, l'authentification souhaitée...
- Redémarrer le client.

---

# Prévoir

- La synchronisation des serveurs
- Le "nettoyage" du "share repository"
- Les différentes mises à jour