

Clonage et déploiement de PC avec Rembo

Pascal Aubry - IFSIC – Université de Rennes 1 - Pascal.Aubry at univ-rennes1.fr

Katy Santerre - IFSIC – Université de Rennes 1 - Katy.Santerre at univ-rennes1.fr

Octobre 2003

Résumé

Après avoir testé plusieurs solutions de clonage sur son réseau, l'IFSIC (UFR d'informatique de l'Université de Rennes 1) a finalement choisi Rembo, un outil puissant de création et de déploiement massif d'images de clients PC.

Fourni avec de nombreux plug-ins et un wizard qui permet d'effectuer toutes les opérations classiques de déploiement d'un parc de PC, Rembo n'est cependant pas nativement dédié à la gestion d'un parc universitaire comme celui de l'IFSIC. Le développement d'un kit (ifsic-rembo) a permis d'adapter Rembo à nos besoins particuliers.

1 Rembo : un environnement de gestion de parcs PC

Rembo [1] est le successeur de *BPBatch* [2][3]. C'est un environnement commercial de création et de déploiement d'images sur PC. Le serveur *Rembo* s'exécute sur *Windows*, *Linux*, *FreeBSD* ou *Sparc Solaris* et s'appuie sur un serveur DHCP. Les clients PC (i386 ou supérieur) doivent supporter PXE [4] (*Pre-eXecution Environment*)¹. Les clients du réseau sont répartis en plusieurs groupes, ce qui permet de définir des comportements par type de machine, par salle, ...

Gestion des images. *Rembo* offre comme tous les outils de clonage la possibilité de récupérer l'image d'un client, la stocker sur le serveur pour ensuite la redistribuer. Le partage de fichiers communs entre plusieurs images permet de réduire la place nécessaire sur le serveur pour le stockage. L'installation/récupération d'une image sur/depuis un client se fait directement ou par différentiel (par rapport à une image de référence).

Exécution pré-OS et machine virtuelle. Au démarrage, les clients récupèrent du serveur TFTP du serveur *Rembo* un environnement d'exécution (machine virtuelle), munie d'un interpréteur de *Rembo-C*, langage de script à la syntaxe proche de *JavaScript*. Il est alors possible, depuis cette machine virtuelle, d'exécuter toutes sortes d'opérations à travers une interface graphique², dont des accès disque de très bas niveau (partitionnement, formatage, installation d'images). Les bibliothèques de *Rembo-C* sont très riches, permettant par exemple l'envoi de courriers électroniques (remontée d'alertes), l'accès à des bases de données via ODBC (pour les *logs* par exemple) ou l'utilisation de variables persistantes (récupérables entre les différents démarrages des clients), propres à un client ou un groupe de clients.

Protocoles réseau. Des protocoles réseau propriétaires, adaptés au mode de stockage des images de *Rembo*, sont utilisés : *NETfs* pour les transferts *unicast*, et *MCAST* pour les transferts *multicast*³. Les transferts entre le serveur *Rembo* et les clients sont chiffrés pour assurer la confidentialité des données.

Cache des images transférées. Comme *BPBatch*, *Rembo* utilise l'espace non partitionné des clients pour stocker les images transférées. Cette gestion de cache permet un gain de performance très impressionnant, ainsi qu'une libération considérable de bande passante sur le réseau puisque les images n'y transitent qu'une seule fois.

Authentification. *Rembo* offre des fonctions d'authentification *Windows* (sur un domaine NT), *Unix* (PAM) et *Radius*. Il permet de donner à des utilisateurs un accès privilégié.

Gestion particulière des systèmes installés. *Rembo* connaît la plupart des systèmes de fichiers couramment utilisés sur PC (FAT, NTFS, ext), et les systèmes d'exploitation correspondant (DOS, *Windows 9x/Me/2000/XP*, *Linux*). Des opérations spécifiques sont prises en charge par *Rembo* : gestion des noyaux sous *Linux*, gestion des SID et accès à la base de registres sous *Windows*.

Tolérance aux fautes. *Rembo* l'assure par redondance du serveur (présence d'un serveur de *backup*), et la reconnexion automatique des clients en cas d'arrêt du serveur principal pendant un transfert.

Outils. L'environnement *Rembo* est muni d'outils d'administration très pratiques : un client en ligne pour accéder aux images du serveur (*NetClnt*), une console graphique pour effectuer des opérations sur le serveur (*Sconsole*), ainsi qu'un outil de prise en main à distance sur les clients (*Rconsole*).

¹ Un support disquette ou CDROM est néanmoins possible pour les clients ne supportant pas PXE.

² L'interface graphique de *Rembo* est basée sur HTML, et la programmation y est événementielle comme JavaScript (onChange...).

³ L'utilisation de *MCAST*, qui utilise des fenêtres « adaptatives » en fonction des données à transférer, montre un gain de performance d'un facteur 5 à 10 par rapport à l'utilisation d'un transfert classique MFTP.

2 *ifsic-rembo* : un kit pour la gestion de parcs PC d'enseignement

Le réseau de l'IFSIC est composé d'environ 400 clients, réparties en salles libre-service et PC personnels des enseignants. À chaque salle libre-service correspond un « groupe *Rembo* », dont tous les membres (les clients de la salle) doivent avoir la même architecture matérielle.

2.1 Utilisation du kit

1a) Préparation d'une image (par l'administrateur). L'administrateur prépare une machine type, configurée « aux petits oignons » pour les utilisateurs. Cette image est ensuite, grâce à l'interface graphique du kit, sauvegardée sur le serveur en quelques clics. À ce moment, l'image créée sur le serveur n'est visible que par les enseignants.

1b) Autorisation de déploiement (par l'administrateur). L'administrateur rend l'image visible pour une ou plusieurs salles. L'image est alors visible par les enseignants, c'est-à-dire qu'elle est « installable » sur les clients.

2) Diffusion (automatique) de l'image sur les clients. Le redémarrage de chaque client est forcé chaque nuit⁴. Lors de ce redémarrage nocturne, les clients téléchargent (si nécessaire, en fonction de ce qu'elles trouvent dans leur cache) toutes les images installables. Celles-ci sont alors immédiatement disponibles en journée. Les systèmes classiques (*Linux*, *Windows*) ne nécessitent qu'une partie seulement de l'espace des disques des PC pour fonctionner (typiquement 4Go pour 40Go disponibles) ; l'espace non utilisé permet de stocker toutes les images installables des clients.

3) Installation d'une image (par les enseignants). Parmi les images installables dans une salle, une est définie par défaut. Lors d'un redémarrage, et après authentification comme enseignant (ou administrateur), il est possible de changer l'image par défaut d'une salle, en quelques clics. Les redémarrages suivants des clients de la salle se feront alors sur cette nouvelle image par défaut.

4) Utilisation courante (utilisateurs). À chaque redémarrage des clients, sans intervention des utilisateurs, les clients sont « réparés » par synchronisation avec l'image par défaut (les fichiers modifiés ou effacés sont remis en place, les fichiers ajoutés sont effacés). Les utilisateurs peuvent également, sur demande, régénérer complètement la machine (repartitionnement, formatage et ré-installation) en cas d'incident majeur.

2.2 Apports du kit

Le kit *ifsic-rembo*, développé en *Rembo-C*, permet de remplir les objectifs suivants :

Facilité de création et de déploiement des images. Le kit *ifsic-rembo* encapsule certaines fonctions complexes de clonage de *Rembo* dans une interface graphique très simple d'utilisation. La prise en main par les administrateurs est très facile.

Stabilité des clients. À chaque démarrage des clients, ceux-ci sont « réparés ». Particulièrement appréciée sur des systèmes réputés peu stables, cette fonctionnalité permet aux utilisateurs de toujours disposer d'un client « neuf ».

Délégation de pouvoirs d'administration. En s'appuyant sur les mécanismes d'authentification offerts par *Rembo*, les utilisateurs sont réparties en trois catégories : les administrateurs (disposant de tous les droits), les enseignants (qui peuvent déployer une image sur une salle), et les étudiants (qui peuvent eux-mêmes régénérer un client).

Limitation de la charge du réseau. Le téléchargement des images par anticipation en période creuse et l'utilisation de l'espace non partitionné en cache fait que les manipulations en période pleine (la journée) n'ont aucune implication sur la charge du réseau.

Versatilité des salles libre-service. Une salle peut, en quelques minutes, changer d'image (passer par exemple de *Linux* à *Windows XP*). Cela permet une très grande souplesse dans la gestion des emplois du temps. De plus cette manipulation est simple (à travers l'interface graphique), peut être faite par les enseignants (sans intervention des administrateurs), et n'a pas d'impact sur la charge du réseau (pré-téléchargement des images).

Le kit *ifsic-rembo* peut être téléchargé sur <http://ifsic-rembo.sourceforge.net>.

Références

[1] Rembo Technology, <http://www.rembo.com>

[2] BPBatch, <http://www.bpbatch.org>

[3] Gestion automatisée de parcs informatiques : BPBatch, Olivier Pagé & Gérard Milhaud, Actes de JRES2001.

[4] Remote Network Boot via PXE, www.kegel.com/linux/pxe.html

⁴ Par connexion à distance (*r-commands*) , par *Wake-On-LAN* ou sur ordre du BIOS.