

# TP CLIENTS LEGERS SOUS THINSTATION \*

François Ducrot - Jacquelin Charbonnel

TP DSI - Juillet 2009 - Angers

## 1 Introduction

*Thinstation* est une petite distribution Linux permettant de booter un PC sans disque par le réseau (via le protocole PXE). Ce PC, (appelé *client léger* ou CL), peut ensuite être utilisé pour se connecter, via divers protocoles, à différents serveurs applicatifs.

La solution *Thinstation* met en oeuvre :

- des PC sans disque destinés à être des CL,
- un serveur d'infrastructure nécessaire au boot des CL, c'est-à-dire assurant essentiellement le rôle de serveur DHCP et TFTP,
- éventuellement, des serveurs applicatifs accessibles par exemple via XDMCP, NX, RDP ou ICA.

Le TP propose 3 manip :

- boot d'un CL avec la distribution *thinstation* de base.
- ajout de clients *standard* (ssh, http, rdp),
- ajout d'un serveur X.

Le serveur d'infrastructure est une machine Fedora 9 (192.168.1.1) sur lequel chacun d'entre vous dispose d'un login, `tpxx`. On s'y connecte par ssh.

Son serveur DHCP est configuré de façon à ce que chaque CL `PCxx` aille chercher ses fichiers dans le répertoire `/tftpboot/tpxx`. Pour plus de détail, voir le fichier de configuration `/etc/dhcpd.conf` du serveur DHCP sur 192.168.1.1.

Le but des manip est que chaque participant `xx` fabrique une arborescence `/tftpboot/tpxx/` comprenant à chaque fois l'ensemble des fichiers nécessaires pour faire démarrer le PC dans la configuration souhaitée.

Tous les fichiers utiles à ce TP se trouvent sur 192.168.1.1 dans le répertoire `/data/thinstation`. Les compilations seront effectuées dans le répertoire `/home/tpxx`, et les fichiers fabriqués seront déposés dans `/tftpboot/tpxx`.

## 2 Manip #1 : boot de la distribution de base

Il s'agit de fabriquer un noyau `vmlinuz` et un disque virtuel `initrd` pour booter le CL.

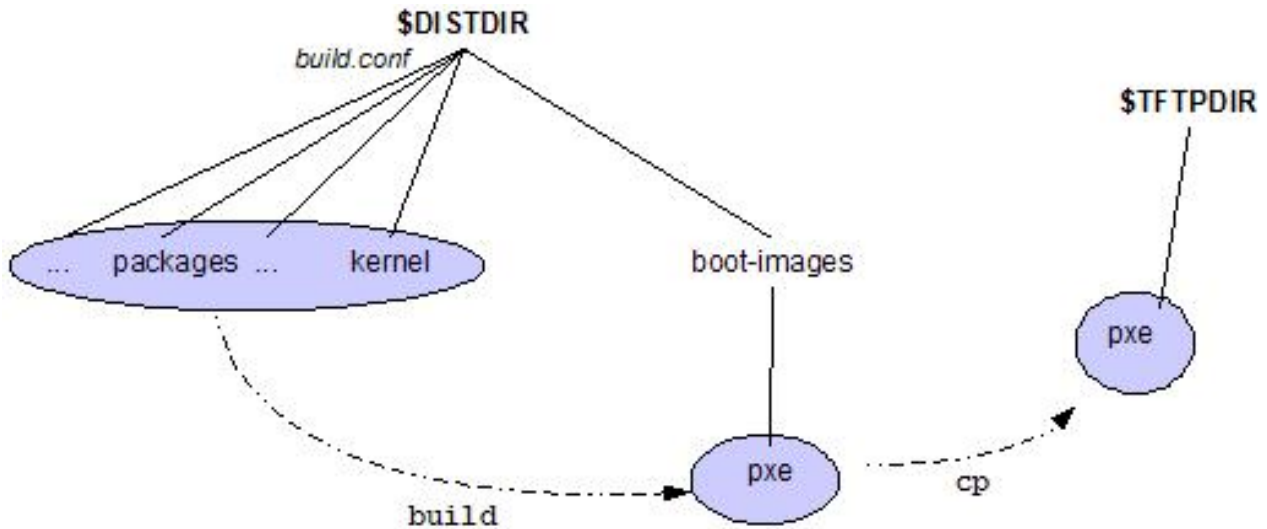
[Q1] ⇒ Sur le serveur 192.168.1.1, décompressez l'archive `/data/thinstation/Thinstation-2.2.2.tar.gz` dans votre home directory (initialisez une variable d'environnement `DISTDIR` pour plus de commodités).

```
$ tar xf /data/thinstation/Thinstation-2.2.2.tar.gz -C ~
$ export DISTDIR=~/.Thinstation-2.2.2
```

[Q2] ⇒ Dans le fichier `$(DISTDIR)/buid.conf`, modifiez la ligne param `basepath` en spécifiant pour valeur `tpxx/config`

[Q3] ⇒ Lancez la construction des fichiers de boot

```
$ (cd $DISTDIR ; ./build)
```



Cela génère des fichiers dans `$DISTDIR/boot-images`, qu'il faut recopier pour les mettre à disposition du serveur TFTP (dans un répertoire `pxe`, pour être conforme à ce qui est spécifié dans la config DHCP) :

[Q4] ⇒ Créez les répertoires dans l'arborescence tftp et copiez les fichiers créés (initialisez une variable TFTPDIR pour faciliter la suite)

```
$ export TFTPDIR=/tftpboot/$USER
$ mkdir -p $TFTPDIR/{pxe,config,pkg}      # config et pkg, c'est pour la suite !
$ cp -av $DISTDIR/boot-images/pxe/{vmlinuz,initrd,pxelinux.{0,cfg}} $TFTPDIR/pxe
```

[Q5] ⇒ Démarrez le CL

Si tout se passe bien, il devrait démarrer (on ignore les messages d'erreur dus à l'absence du fichier principal de configuration). Vous devriez donc vous retrouver devant un gestionnaire de fenêtre (`icewm`). Vous pouvez déjà lancer un `xterm` et examiner le contenu du CL.

Que s'est-il passé ?

- le CL a chargé par TFTP `pxelinux.0`
- `pxelinux` a chargé un noyau `vmlinuz` et un disque virtuel `initrd`
- le CL a démarré sur ce noyau/disque virtuel, et a essayé de récupérer des fichiers de configuration : ici, il n'y en avait pas.

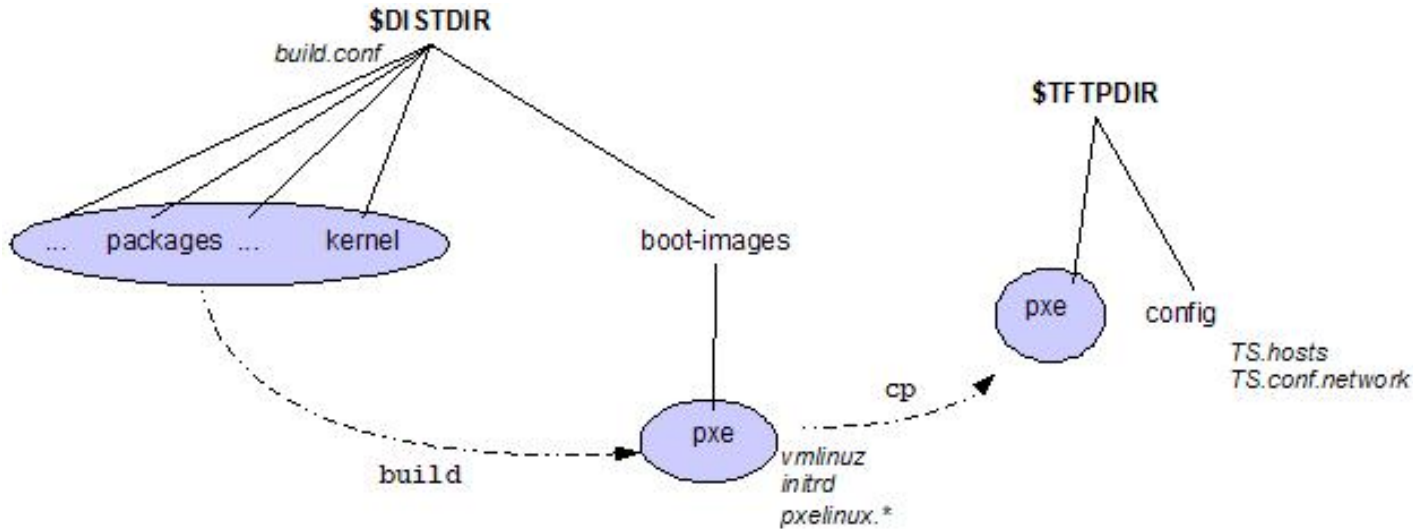
### 3 Manip #2 : personnalisation d'un parc de CL

Des fichiers de configuration permettent de personnaliser individuellement un parc de CL.

Suivant le fichier dans lequel il est défini, un paramètre sera commun à tous les CL, spécifique à un CL donné, ou spécifique à un groupe de CL. Les paramètres peuvent être spécifiés dans n'importe quel fichier, et on peut créer autant de fichiers de conf que l'on veut.

Les fichiers sont mis à disposition sur le serveur TFTP (dans le répertoire spécifié par le paramètre `basepath`, initialisé lors de la manip #1), et qui sont récupérés par les CL au boot.

On peut par exemple créer un fichier `TS.conf-xxxx` pour paramétrer un CL particulier, ou créer un fichier `TS.conf.group-xxxx` pour paramétrer un ensemble de CL.



Plus précisément :

- `TS.hosts`, s'il est présent, associe à chaque adresse MAC un nom de machine (pas forcément celui du DNS) et un certain nombre de groupes auxquels appartient ce CL. Chaque CL est référencé par une ligne de la forme :

```
tp01 001ec9360931 hres x11 nx
```

qui signifie que la machine `tp01` possède l'adresse MAC `001ec9360931` et appartient aux groupes `hres`, `x11` et `nx`.

- `TS.conf.network`, s'il existe, précise des paramètres communs à tous les CL
- `TS.conf-tp01` : paramètres pour la machine `tp01` (où `tp01` est un nom de machine défini dans `TS.hosts`)
- `TS.conf-001ec9360931` ou `TS.conf-192.168.1.11` : paramètres pour une machine d'adresse MAC ou d'adresse IP donnée
- `TS.conf.group-hres` paramètres pour le groupe de machines `hres` (où `hres` est un groupe de CL défini dans `TS.hosts`)

Dans ce qui suit, on va créer `TS.hosts` et un fichier de groupe `TS.conf.group-hres` pour les CL ayant un écran haute résolution :

1. Configuration du groupe

[Q6] ⇒ Créez un fichier `$TFTPDIR/config/TS.conf.group-hres` contenant la ligne :

```
SCREEN_RESOLUTION=1280x1024
```

2. Création du fichier `TS.hosts`

On crée un fichier `TS.hosts` indiquant que le CL appartient au groupe `hres`.

[Q7] ⇒ Récupérez l'adresse MAC de votre CL et créez un fichier `$TFTPDIR/config/TS.hosts`.

```
$ grep $USER /etc/dhcpd.conf
$ echo "$USER adresse-mac-aabbccddeeff hres" > $TFTPDIR/config/TS.hosts
```

[Q8] ⇒ Rebootez le CL et testez.

Vous pouvez lancer une session `rdesktop` en cliquant sur l'icône windows, et en rentrant comme serveur `172.19.45.12`.

## 4 Manip #3 : ajout d'un serveur X

Dans ce qui suit, on va créer un nouveau groupe pour les CL que l'on veut doter d'un serveur X.

[Q9] ⇒ Créez un fichier `$TFTPDIR/config/TS.conf.group-X11` contenant les lignes suivantes :

```
SESSION_1_TYPE=x
SESSION_1_SCREEN=1
SESSION_1_X_SERVER=192.168.1.1
SESSION_1_X_OPTIONS="-query"

SESSION_2_TYPE=x
SESSION_2_SCREEN=2
SESSION_2_X_SERVER=192.168.1.2
SESSION_2_X_OPTIONS="-query"
```

[Q10] ⇒ **Ajoutez votre CL au groupe X11 en ajoutant X11 à la ligne du fichier \$TFTPDIR/config/TS.hosts**

A ce niveau, le CL doit pouvoir ouvrir une session X11 sur le serveur interactif.

[Q11] ⇒ **Redémarrez le CL et testez**

Que s'est-il passé? En fin de boot, le CL a récupéré le fichier `TS.hosts`, en a déduit qu'il fallait ensuite charger `TS.conf.group-hres` et `TS.conf.group.X11`, ce qui l'a conduit à ouvrir une session XDM sur `192.168.1.1`.

Dans cette config, le CL peut se connecter simultanément à 2 serveurs interactifs.

[Q12] ⇒ **Jouez avec les touches Ctrl Alt F{1,2,3,4,5}**

**Conclusion** - La mise au point d'une telle architecture prend un certain temps. Mais une fois opérationnelle, l'installation d'un nouveau poste de travail se résume à :

- sortir le PC du carton
- retirer le disque dur (optionnel)
- brancher le secteur, le réseau et récupérer l'adresse MAC
- ajouter une entrée au serveur DHCP et une entrée au fichier `TS.hosts`
- booter

soit 5 mn maximum. Et zéro intervention sur le poste utilisateur par la suite...