Mini-HOWTO LBX

Paul D. Smith, *psmith@baynetworks.com* , trad. : *Xavier Glattard* v1.04, 11 Décembre 1997, trad. Juin 1997

LBX (Low Bandwidth X, X Faible Bande-passante) est une extension du serveur X qui effectue une compression du protocole X. En d'autres termes, elle permet, dans le cas d'applications X et d'un serveur X séparés par une connexion réseau lente, d'accélérer l'affichage et de réduire les temps de réponse.

Contents

1 Introduction

Low-Bandwidth X (LBX) prend en compte le fait que de nos jours tout le monde ne travaille pas qu'à une ou deux connexions rapides du serveur sur lequel tournent ses applications.

Le protocole X peut générer un trafic extraordinaire sur votre réseau, en particulier pour des choses apparemment simples telles que la création de nouvelles fenêtres. Quiconque aura essayé d'utiliser X à travers une connexion modem à 28,8 (ou même mieux) vous le dira : créer de nouvelles fenêtres X nécessite une patience à toute épreuve.

Le principe est le suivant : LBX est un système de compression et de cache conçu pour minimiser le trafic X généré entre deux systèmes.

2 Quel est le statut de LBX ?

En tant que composant de la publication X11R6.3 du Consortium X datant de décembre 1996, LBX est une authentique extension du protocole X. Pour les utilisateurs de XFree86, cela correspond à XFree86 version 3.3.

3 Pourquoi utiliser LBX ?

Si vous utilisez un modem pour vous connecter à un prestataire de service, et que vous exécutez des applications X sur des machines distantes, l'affichage (variables DISPLAYs) étant dirigé sur votre machine (et vice versa), alors LBX accélérera la connexion. De même, si vous redirigez l'affichage d'autres systèmes à travers des réseaux étendus (d'autres pays, par exemple) ou à travers des connexions lentes, LBX peut vous être utile.

4 Pourquoi ne pas utiliser LBX ?

LBX est inutile, naturellement, si vous exécuter des applications localement, ou si vous n'utilisez pas X.

De même, si vous travaillez sur un réseau local rapide, LBX ne vous sera pas d'une grande utilité. Certains disent : "si LBX réduit le trafic réseau, ne pourrait-on pas l'utiliser sur des réseaux locaux rapides ?" On pourrait, si le but est de réduire le trafic sur le réseau. Néanmoins, cela introduirait du cache et de la compression, qui consomment des ressources à chaque extrémités (de la mémoire supplémentaire pour le cache, et du temps CPU pour la décompression). Si votre liaison est plutôt performante, LBX causera sans doute un ralentissement global du réseau.

5 Comment fonctionne LBX ?

LBX fonctionne en introduisant un *serveur proxy* du côté du client, lequel se charge du cache et de la compression. Le serveur X sait que le client utilise un serveur proxy, et décompresse en conséquence.

Voici un exemple de configuration classique pour des clients X distants. Dans la suite, LOCAL représentera toujours la station qui se trouve en face de vous, et dont vous regardez le moniteur, et DISTANT sera la station distante, sur laquelle votre application est exécutée.

```
   DISTANT                                      LOCAL
+-----+                                                  +-----+
| APP |-\           Reseau           +-----------+  |     |\
+-----+  \-------------------------->| SERVEUR X |=>|     ||
+-----+  /     (Protocole X)         +-----------+  +-----+\
| APP |-/                                          /_____//
+-----+
```

Lors de l'utilisation de LBX, un serveur proxy (`lbxproxy`) est introduit du côté distant, et les applications communiquent avec ce processus et non plus directement avec le serveur LOCAL. Ce processus se charge alors du cache et de la compression des requêtes X et les transmet. Cela ressemble à ça :

```
          DISTANT                                        LOCAL
                                                                  +-----+
      +-----+  +-------+        Reseau          +-----------+  |     |\
      | APP |->| PROXY |------------------------>| SERVEUR X |=>|     ||
      +-----+  +-------+    (LBX/Protocole X)    +-----------+  +-----+\
      +-----+   /                                                /_____//
      | APP |--/
      +-----+
```

Le détail des fonctions de cache et de compression de LBX sort du cadre de ce document.

6 De quoi ai-je besoin pour utiliser LBX ?

Vous avez besoin sur votre système LOCAL d'un serveur X compilé avec l'extension LBX. A moins que vous n'ayez explicitement demandé le contraire, les serveurs X11R6.3 incluent automatiquement LBX à la compilation. Par conséquent, tous les serveurs XFree86 3.3 incluent LBX par défaut.

Vous pouvez utiliser la commande `xdpyinfo` afin de vérifier si votre serveur dispose de l'extension LBX : exécutez `xdpyinfo` et vérifiez la liste qui suit "number of extensions" (nombre d'extensions); vous devez voir "LBX" dans cette liste.

Ensuite, vous avez besoin d'un programme `lbxproxy` compilé pour le système DISTANT. C'est le point le plus délicat. Si le système distant n'est pas du même type que votre système local, le `lbxproxy` de votre système local ne sera pas bon, évidement.

Malheureusement, aucune distribution de `lbxproxy` n'a jamais été publiée. Par conséquent, vous devez soit (a) obtenir et compiler la majeure partie, sinon la totalité, de X11R6.3 pour le système distant, soit (b) trouver quelque part un exécutable `lbxproxy` précompilé pour votre système. Cette dernière solution est naturellement la plus simple.

`lbxproxy` est constitué d'un unique fichier exécutable. Aucun fichier de configuration, de ressources, etc. ne lui est associé.

7 De quoi n'ai-je pas besoin pour utiliser LBX ?

Le système DISTANT **n'a pas** besoin d'un nouveau serveur X (dans tous les cas, le système DISTANT n'a besoin d'exécuter *aucun* serveur X).

L'application que vous voulez exécuter **n'a pas** besoin d'être liée à une version spéciale de X, ou à une bibliothèque spéciale; j'utilise régulièrement des applications commerciales X11R5 à travers LBX sans aucun souci.

Vous **n'avez pas** besoin de droits d'accès "root" ou privilégiés sur le système DISTANT; le processus `lbxproxy` utilise vos droits d'accès normaux. De plus, vous pouvez l'exécuter depuis votre répertoire personnel : il n'a pas besoin d'être installé à un endroit particulier.

8 Comment je démarre LBX ?

Bon, nous y voilà... après tout, rien de bien compliqué jusqu'à présent. Remplacez dans la suite LOCAL et DISTANT respectivement par les noms d'hôte de votre station locale et du système distant (ne mélangez pas tout!).

Sur LOCAL :

1. Démarrez votre serveur X.
2. Donnez les droits d'accès du système distant auprès de votre serveur X. Avec la méthode de la liste d'hôtes, tapez `xhost +DISTANT`. Si vous utilisez `xauth` cela risque de ne pas être suffisant; consultez le manuel *xauth(1)* pour plus d'informations.

   Vous pourriez consulter efficacement le mini howto *Remote X Apps* `http://www.freenix.fr/linux/HOWTO/mini/Remote-X-Apps.html`

   si vous n'êtes pas familier avec la configuration des droits d'accès distants sous X.

Sur DISTANT :

1. Démarrez `lbxproxy` en précisant la redirection vers le serveur X LOCAL, comme cela :

   ```
   $ lbxproxy -display LOCAL :0 :1 &
   ```

   Cette commande indique à `lbxproxy` d'utiliser l'écran ("display") `:1` sur le système DISTANT; si ce système dispose déjà de plus d'un écran, vous pouvez choisir `:2`, ou n'importe quoi d'autre.
2. Définissez votre variable d'environnement DISPLAY afin qu'elle pointe vers l'écran géré par `lbxproxy`, au lieu de l'écran habituel :

   ```
   $ DISPLAY= :1
   $ export DISPLAY
   ```

   Ou, si vous utilisez csh ou un de ses clones :

   ```
   % setenv DISPLAY :1
   ```
3. Si vous utilisez `xauth`, vous aurez à vérifier que votre "cookie" est accessible localement. Consultez le mini howto *Remote X Apps* `http://www.freenix.fr/linux/HOWTO/mini/Remote-X-Apps.html` pour plus d'informations à ce propos.
4. Démarrez vos applications X!

Voilà; toute application démarrée vers l'écran `:1` utilisera LBX. Naturellement, il n'y a aucune raison pour que vous ne puissiez pas également démarrer des applis X vers `LOCAL :0` et les utiliser simultanément.

9 Problèmes

Voici quelques problèmes courants :

Q)

`lbxproxy` se termine avec l'erreur "access denied" ("accès refusé").

R)

Cela signifie que le système LOCAL n'accepte pas les connexions en provenance du système DISTANT pour des raisons d'autorisation. Consultez le mini howto *Remote X Apps* `http://www.freenix.fr/linux/HOWTO/mini/Remote-X-Apps.html` pour plus de détails à ce sujet.

En guise de test simple, essayez de lancer sur DISTANT une appli simple, comme `xclock`, en l'affichant sur le système local, sans utiliser `lbxproxy` :

```
$ xclock -display LOCAL :0
```

Si cela ne marche pas, le problème vient de `xhost`, ou d'une simple anomalie de X, pas de LBX.

10 Documentation

La seule documentation fournie avec une distribution X standard est probablement la page de manuel de *lbxproxy(1)*.

Si vous pouvez consulter l'arborescence des sources de X, de très intéressantes informations sont disponibles là :

- `xc/doc/specs/Xext/lbx.mif` (Framemaker MIF)
- `xc/doc/hardcopy/Xext/lbx.PS.Z` (Postscript compressé)
- `xc/doc/hardcopy/Xext/lbxTOC.html` (HTML)

Une discussion plus précise à propos des algorithmes spécifiques à LBX est disponible là :

- `xc/doc/specs/Xext/lbxalg.mif` (Framemaker MIF)
- `xc/doc/specs/Xext/lbxalg.PS.Z` (Postscript compressé)

Si vous n'avez pas accès à l'arborescence des sources de X11, vous pouvez obtenir ces fichiers depuis *le site FTP du Consortium X* `ftp://ftp.x.org/pub/R6.3/xc/doc/` .

11 Alternatives

Si, pour quelque raison, vous n'aimez pas `lbxproxy` : vous n'êtes pas satisfait des performances, ou bien ça ne marche pas pour vous, ou vous ne voulez pas vous casser la tête à créer un lbxproxy pour le système distant, ou encore vous avez tout simplement envie d'essayer d'autres solutions, alors il existe au moins un autre kit de compression du protocole X (quelqu'un en connaît d'autres?)

11.1 dxpc - Differential X Protocol Compressor

- Auteur initial : *Brian Pane <brianp@cnet.com>*
- Actuel responsable : *Zachary Vonler <lightborn@mail.utexas.edu>*

dxpc `http://ccwf.cc.utexas.edu/~zvonler/dxpc/` (Differential X Protocol Compressor, Compresseur Différentiel de protocole X) fonctionne pour l'essentiel de la même manière que LBX. Cependant, afin d'éviter d'avoir à implémenter une extension X et à modifier le code du serveur X, `dxpc` utilise **deux** proxys : le premier s'exécute sur le système DISTANT, comme `lbxproxy`, et l'autre s'exécute sur l'hôte LOCAL.

Le proxy de l'hôte DISTANT intervient dans la communication entre les clients X et le proxy de l'hôte LOCAL, tandis que le proxy de l'hôte LOCAL intervient dans la communication entre le serveur X et le proxy de l'hôte DISTANT.

Ainsi, à la fois pour les clients X et pour le serveur X, cela ressemble à une connexion X normale.

11.1.1 Avantages

- Comme il s'agit d'une application complètement indépendante qui n'utilise aucune particularité de X, elle est plus simple à compiler et à installer.
- Elle est maintenue séparément, aussi vous n'avez pas à attendre la publication par l'OSF de nouvelles versions de X pour profiter d'améliorations ou de corrections.
- Elle fournie des informations et des statistiques plus nombreuses et plus complètes que `lbxproxy`.

11.1.2 Inconvénients

- Ce n'est pas un composant standard de X; vous devez l'obtenir et l'installer séparément.
- dxpc est légèrement plus complexe à configurer, puisqu'un proxy est nécessaire sur LOCAL, en plus du proxy DISTANT.

11.1.3 Où obtenir dxpc?

Le code source de dxpc est disponible à *ftp.x.org* `ftp://ftp.x.org/contrib/utilities/` .

Une page web sur dxpc donne beaucoup d'informations intéressantes, y compris des liens vers la liste de diffusion dxpc, un accès au code source, et un certain nombre d'exécutables précompilés pour différentes plates-formes :

`http://ccwf.cc.utexas.edu/~zvonler/dxpc/`

11.2 Ssh (Secure Shell)

Ken Chase <lbxhowto@sizone.org>

précise que ***ssh*** `http://www.cs.hut.fi/ssh/` peut être utilisé pour la compression. Bien que sa principale fonction soit de sécuriser, il compresse également les données qu'il envoie.

Ainsi, si vous utilisez X à travers une connexion `ssh`, vous obtiendrez automatiquement un certain taux de compression.

11.3 Lequel choisir ?

Je ne sais pas. LBX et `dxpc` apportent certainement tous les deux une meilleure compression que `ssh`. Bien sûr, `ssh` a pour lui l'avantage de la sécurisation. Et bien sûr, il n'y a aucune raison pour que vous ne puissiez pas utiliser à la fois `ssh` et l'un des deux autres, afin d'obtenir une bonne compression et une sécurisation.

Il ne devrait pas être très difficile de réaliser un test comparatif de ces différentes solutions, afin de disposer de mesures statistiques et subjectives des performances. Mais je ne l'ai pas fait, et je ne connais personne qui l'ait fait.