

Technologie NX (FreeNX/Nomachine)

Sylvain Ferrand

Ecole Polytechnique

19 octobre 2009, Journées Mathrice, Université d'Angers



Qu'est ce que c'est

Un protocole pour des connexions graphiques distantes

- Permettant des connexions sur des liens à faible bande passante ou à forte latence
- Sécurisé par SSH

Avec comme autres avantages

- persistance de session (à la screen)
- proxifier des connexions VNC et RDP
- installation multiplateforme triviale
- impression locale et le partage de fichiers



Qu'est ce que c'est

Un protocole pour des connexions graphiques distantes

- Permettant des connexions sur des liens à faible bande passante ou à forte latence
- Sécurisé par SSH

Avec comme autres avantages

- persistance de session (à la screen)
- proxifier des connexions VNC et RDP
- installation multiplateforme triviale
- impression locale et le partage de fichiers



Qu'est ce que c'est

Un protocole pour des connexions graphiques distantes

- Permettant des connexions sur des liens à faible bande passante ou à forte latence
- Sécurisé par SSH

Avec comme autres avantages

- persistance de session (à la screen)
- proxifier des connexions VNC et RDP
- installation multiplateforme triviale
- impression locale et le partage de fichiers



Qu'est ce que c'est

Un protocole pour des connexions graphiques distantes

- Permettant des connexions sur des liens à faible bande passante ou à forte latence
- Sécurisé par SSH

Avec comme autres avantages

- persistance de session (à la screen)
- proxifier des connexions VNC et RDP
- installation multiplateforme triviale
- impression locale et le partage de fichiers



Qu'est ce que c'est

Un protocole pour des connexions graphiques distantes

- Permettant des connexions sur des liens à faible bande passante ou à forte latence
- Sécurisé par SSH

Avec comme autres avantages

- persistance de session (à la screen)
- proxifier des connexions VNC et RDP
- installation multiplateforme triviale
- impression locale et le partage de fichiers



Qu'est ce que c'est

Un protocole pour des connexions graphiques distantes

- Permettant des connexions sur des liens à faible bande passante ou à forte latence
- Sécurisé par SSH

Avec comme autres avantages

- persistance de session (à la screen)
- proxifier des connexions VNC et RDP
- installation multiplateforme triviale
- impression locale et le partage de fichiers



Qu'est ce que c'est

Un protocole pour des connexions graphiques distantes

- Permettant des connexions sur des liens à faible bande passante ou à forte latence
- Sécurisé par SSH

Avec comme autres avantages

- persistance de session (à la screen)
- proxifier des connexions VNC et RDP
- installation multiplateforme triviale
- impression locale et le partage de fichiers



Comment ça marche?

Le serveur NX compresses les données X11 pour réduire la bande passante

- Utilisation d'un cache (exemple - les menus)
- compression des images (avec ou sans perte)
- adaptation des méthodes de compression en fonction de la bande passante
- réduction du nombre d'aller/retour de messages X (requête/réponse)

Une connexion réellement utilisable à travers un modem.
Incomparable par rapport à (ssh -C -X)



Comment ça marche?

Le serveur NX compresses les données X11 pour réduire la bande passante

- Utilisation d'un cache (exemple - les menus)
- compression des images (avec ou sans perte)
- adaptation des méthodes de compression en fonction de la bande passante
- réduction du nombre d'aller/retour de messages X (requête/réponse)

Une connexion réellement utilisable à travers un modem.
Incomparable par rapport à (ssh -C -X)



Comment ça marche?

Le serveur NX compresses les données X11 pour réduire la bande passante

- Utilisation d'un cache (exemple - les menus)
- compression des images (avec ou sans perte)
- adaptation des méthodes de compression en fonction de la bande passante
- réduction du nombre d'aller/retour de messages X (requête/réponse)

Une connexion réellement utilisable à travers un modem.
Incomparable par rapport à (ssh -C -X)



Comment ça marche?

Le serveur NX compresse les données X11 pour réduire la bande passante

- Utilisation d'un cache (exemple - les menus)
- compression des images (avec ou sans perte)
- adaptation des méthodes de compression en fonction de la bande passante
- réduction du nombre d'aller/retour de messages X (requête/réponse)

Une connexion réellement utilisable à travers un modem.
Incomparable par rapport à (ssh -C -X)



Comment ça marche?

Le serveur NX compresse les données X11 pour réduire la bande passante

- Utilisation d'un cache (exemple - les menus)
- compression des images (avec ou sans perte)
- adaptation des méthodes de compression en fonction de la bande passante
- réduction du nombre d'aller/retour de messages X (requête/réponse)

Une connexion réellement utilisable à travers un modem.
Incomparable par rapport à (ssh -C -X)



Comment ça marche?

Le serveur NX compresses les données X11 pour réduire la bande passante

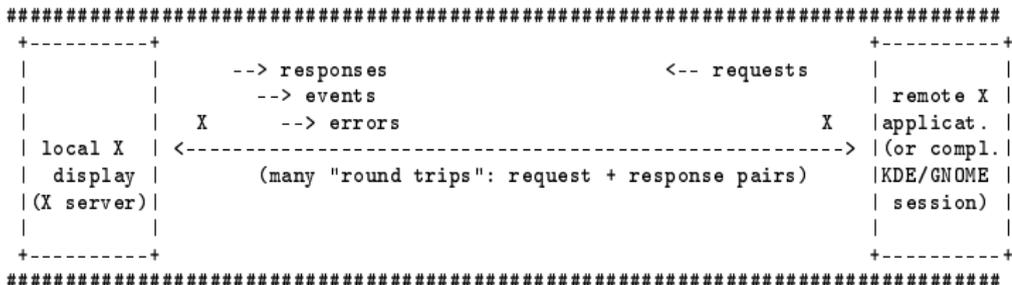
- Utilisation d'un cache (exemple - les menus)
- compression des images (avec ou sans perte)
- adaptation des méthodes de compression en fonction de la bande passante
- réduction du nombre d'aller/retour de messages X (requête/réponse)

Une connexion réellement utilisable à travers un modem.
Incomparable par rapport à (ssh -C -X)

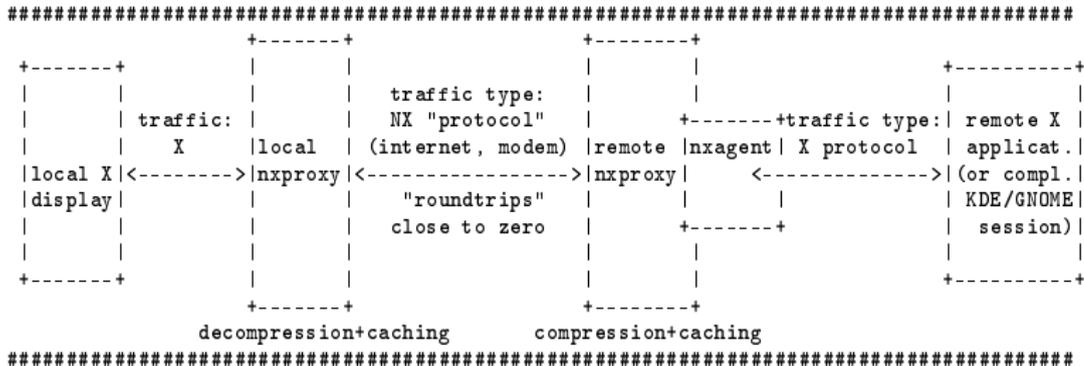


Architecture

X:



NX:



(c) Kurt Pfeifle, Danka Deutschland GmbH <kpfeifle at danka dot de>



Nomachine ou FreeNX

- La société Nomachine développe le protocole NX
 - publie les principales bibliothèques en GPL
 - propose un serveur payant et une version gratuite limitée (NXserver free)
 - fournit des clients binaires mais gratuit pour Windows, linux et MacOS
 - propose un support commercial
- le projet libre FreeNX
 - Propose une implémentation libre du serveur utilisant les bibliothèques de Nomachine



Tout passe par SSH.

Authentification en deux étapes

- Connexion avec une clé publique avec l'utilisateur `nx`
- Authentification par mot de passe sur le système ou par une base de données de mots de passe



Tout passe par SSH.

Authentification en deux étapes

- Connexion avec une clé publique avec l'utilisateur `nx`
- Authentification par mot de passe sur le système ou par une base de données de mots de passe



Sécurité

Tout passe par SSH.

Authentification en deux étapes

- Connexion avec une clé publique avec l'utilisateur `nx`
- Authentification par mot de passe sur le système ou par une base de données de mots de passe



Installation client

Coté client:

- Client propriétaire gratuit nomachine pour Windows, linux et MacOS X
- Client libre QtNX pour linux
- Client web (NX web companion) Java+plugin, propriétaire

Le client propriétaire a une petite longueur d'avance.

Dans 95% des cas l'installation se fait en moins de deux minutes.

A savoir:

Sous windows attention aux conflits avec les bibliothèques cygwin.

Sous Ubuntu attention à apparmor qui n'apprécie pas que l'utilisateur lance un usermode cupsd.



Installation client

Coté client:

- Client propriétaire gratuit nomachine pour Windows, linux et MacOS X
- Client libre QtNX pour linux
- Client web (NX web companion) Java+plugin, propriétaire

Le client propriétaire a une petite longueur d'avance.

Dans 95% des cas l'installation se fait en moins de deux minutes.

A savoir:

Sous windows attention aux conflits avec les bibliothèques cygwin.

Sous Ubuntu attention à apparmor qui n'apprécie pas que l'utilisateur lance un usermode cupsd.



Installation client

Coté client:

- Client propriétaire gratuit nomachine pour Windows, linux et MacOS X
- Client libre QtNX pour linux
- Client web (NX web companion) Java+plugin, propriétaire

Le client propriétaire a une petite longueur d'avance.

Dans 95% des cas l'installation se fait en moins de deux minutes.

A savoir:

Sous windows attention aux conflits avec les bibliothèques cygwin.

Sous Ubuntu attention à apparmor qui n'apprécie pas que l'utilisateur lance un usermode cupsd.



Installation client

Coté client:

- Client propriétaire gratuit nomachine pour Windows, linux et MacOS X
- Client libre QtNX pour linux
- Client web (NX web companion) Java+plugin, propriétaire

Le client propriétaire a une petite longueur d'avance.

Dans 95% des cas l'installation se fait en moins de deux minutes.

A savoir:

Sous windows attention aux conflits avec les bibliothèques cygwin.

Sous Ubuntu attention à apparmor qui n'apprécie pas que l'utilisateur lance un usermode cupsd.



Installation client

Coté client:

- Client propriétaire gratuit nomachine pour Windows, linux et MacOS X
- Client libre QtNX pour linux
- Client web (NX web companion) Java+plugin, propriétaire

Le client propriétaire a une petite longueur d'avance.

Dans 95% des cas l'installation se fait en moins de deux minutes.

A savoir:

Sous windows attention aux conflits avec les bibliothèques cygwin.

Sous Ubuntu attention à `apparmor` qui n'apprécie pas que l'utilisateur lance un `usermode cupsd`.



Installation serveur

Les versions prépackagées s'installent facilement mais sont rarement dans les dépôts officiels

- Ubuntu - Pas dans les dépôts officiels mais paquets maintenus par la FreeNX Team PPA
- CentOS/RHEL - Packets FreeNX 0.7.3 dans extra
- Debian - Utiliser les packets Ubuntu de www.datakeylive.com/ubuntu pour FreeNX 0.7.2 i386 et AMD64



Impression locale

Permet d'imprimer facilement sur une imprimante du client.

- l'utilisateur choisit les imprimantes à partager dans `nxclient`
- `nxclient` lance un `cupsd` local avec les droits de l'utilisateur.
- NX lance un `cups` sur le server avec les droits de l'utilisateur et crée l'imprimante qui imprime sur le port 3000
- NX forward le port 3000 du server vers le cups du client à travers le tunnel
- Position la variable `CUPS_SERVER` dans l'environnement utilisateur pour que les applications utilisent le cups userland



Impression locale

Permet d'imprimer facilement sur une imprimante du client.

- l'utilisateur choisit les imprimantes à partager dans `nxclient`
- `nxclient` lance un `cupsd` local avec les droits de l'utilisateur.
- NX lance un `cups` sur le server avec les droits de l'utilisateur et crée l'imprimante qui imprime sur le port 3000
- NX forward le port 3000 du server vers le cups du client à travers le tunnel
- Position la variable `CUPS_SERVER` dans l'environnement utilisateur pour que les applications utilisent le cups userland



Impression locale

Permet d'imprimer facilement sur une imprimante du client.

- l'utilisateur choisit les imprimantes à partager dans `nxclient`
- `nxclient` lance un `cupsd` local avec les droits de l'utilisateur.
- NX lance un `cups` sur le server avec les droits de l'utilisateur et crée l'imprimante qui imprime sur le port 3000
- NX forward le port 3000 du server vers le cups du client à travers le tunnel
- Position la variable `CUPS_SERVER` dans l'environnement utilisateur pour que les applications utilisent le cups userland



Impression locale

Permet d'imprimer facilement sur une imprimante du client.

- l'utilisateur choisit les imprimantes à partager dans `nxclient`
- `nxclient` lance un `cupsd` local avec les droits de l'utilisateur.
- NX lance un `cups` sur le server avec les droits de l'utilisateur et crée l'imprimante qui imprime sur le port 3000
- NX forward le port 3000 du server vers le cups du client à travers le tunnel
- Position la variable `CUPS_SERVER` dans l'environnement utilisateur pour que les applications utilisent le cups userland



Impression locale

Permet d'imprimer facilement sur une imprimante du client.

- l'utilisateur choisit les imprimantes à partager dans `nxclient`
- `nxclient` lance un `cupsd` local avec les droits de l'utilisateur.
- NX lance un `cups` sur le server avec les droits de l'utilisateur et crée l'imprimante qui imprime sur le port 3000
- NX forward le port 3000 du server vers le cups du client à travers le tunnel
- Position la variable `CUPS_SERVER` dans l'environnement utilisateur pour que les applications utilisent le cups userland



FreeNX comme proxy RDP et RFB

FreeNX peut-être utilisé comme proxy pour les connexions vers des serveurs RDP ou VNC.

Offre aux connexions:

- Sécurité et isolation des machines - grace au tunnel SSH
- Compression - grace à des modules spécifiques aux protocoles RDP et RFB.

La mise en oeuvre est triviale, il faut indiquer la machine de destination via le client.



FreeNX comme proxy RDP et RFB

FreeNX peut-être utilisé comme proxy pour les connexions vers des serveurs RDP ou VNC.

Offre aux connexions:

- Sécurité et isolation des machines - grace au tunnel SSH
- Compression - grace à des modules spécifiques aux protocoles RDP et RFB.

La mise en oeuvre est triviale, il faut indiquer la machine de destination via le client.



Autres logiciels comparables

- XDMCP (X11)
- VLC (RFB) et ses variantes
- 2xTerminal server, Sun secure global Desktop, CITRIX ICA, MS Terminal Server...
- x2go

Pour comparer:

http://en.wikipedia.org/wiki/Comparison_of_remote_desktop_software



Critiques

FreeNX souffre de défauts de conception et les développeurs ont entamé un redesign complet.

Un autre projet basé sur les mêmes bibliothèques: x2go mais les clients ne sont pas compatibles. 2xgo fournit des clients Qt et GTK multiplateforme et GPL.



Références

- <http://freenx.berlios.de>
- <http://www.nomachine.com>
- Série d'article dans Linux Journal
<http://www.linuxjournal.com/article/8477>
- www.x2go.org

