NFS et Kerberos (c'est dans les vieux pots qu'on fait les meilleures soupes mais parfois elles sont salées)

Jean-Marc Pouchoulon

Février 2021

1 Objectifs du TP et organisation.

1.1 Les compétences à acquérir à la fin de cette séance sont les suivantes :

— Installer et utiliser NFS 3 et 4.
— Installer des outils de mesures de la performance d'un système de fichier.
— Testez les performances de NFS dans un scénario "IO bound".
— Comparer NFS avec ext4.
— Sécuriser SSH et NFS via Kerberos

1.2 Organisation, recommandations et notation du TP.

Ce TP a été préparé sur une machine virtuelle Debian. Vous allez travailler par groupes de deux machines virtuelles avec un client et un serveur NFS. Les TP sont donc à rendre par groupe de deux. Le but de ce TP est d'appréhender NFS dans ces quatre aspects système, sécurité, réseaux et performance. Il vous est demandé lors de chaque installation de faire varier certains paramètres pour voir s'ils ont une incidence sur les performances.

Pour travailler avec NFS, prêtez attention à la résolution de nom et à la mise à l'heure des machines via NTP.

Il n'y a pas de travail sans preuves du travail : Vous devez donner des copies d'écrans montrant que vous avez correctement configuré vos services ainsi que vos configurations avec des explications.

2 Installation de NFS

2.1 pré-requis

Installez ntp sur le client et le serveur par exemple en utilisant celui de systemd.

```
systemctl start systemd-timesyncd
systemctl enable systemd-timesyncd
```

2.1.1 Installation du DNS resolver unbound sur nfsserveur.iutbeziers.fr

Récupérez et jouez le script d'installation install_unbound.sh :

```
git clone https://registry.iutbeziers.fr:11443/pouchou/conf-unbound.git
conf-unbound/install_unbound.sh
```

Adaptez les adresses IP dans le fichier /etc/unbound/unbound.conf.d/iutb-unbound.conf avec votre client et votre serveur NFS. Pensez à faire écouter unbound sur votre loopback et sur l'adresse de votre nfsserveur (rajouter une ligne interface).

Finalisez :

```
systemctl restart unbound
systemctl enable unconf
```

Il faut que la résolution de nom fonctionne (host et reverse).Vous devez installer le resolver unbound pour avoir un dns local à vos machines. C'est indispensable pour NFS Kerberos. Modifiez /etc/systemd/-resolved.conf :

— en rajoutant la ligne DNS=127.0.0.1 pour forwardez systemd-resolved vers unbound

— en rajoutant la ligne LLMNR=no pour éviter l'utilisation de la resolution DNS par multicast [1]

Redémarrez systemd-resolved

```
systemctl restart systemd-resolved
```

La résolution se fait désormais depuis la 127.0.0.53 (systemd-resolved) qui "forwarde" vers la 127.0.0.1 (Unbound) qui "forwarde" ensuite vers la 10.255.255.200 (DNS bind de l'IUT).

Unbound vous permet de contrôler votre environnement DNS en rajoutant des enregistrements.

Vérifiez **impérativement** [2] via la commande dig que la résolution et la résolution inverse (dig -x adresse_ip) est fonctionnelle pour chaque machine.

2.2 Installation du serveur NFS

```
apt-get install nfs-kernel-server nfs4-acl-tools nfswatch
mkdir –m 1777 /exportnfs4
mkdir –m 1777 /exportnfs4/users
mkdir –m 1777 /home/users
mkdir -m 1777 /exportnfs3
```

Le fichier /etc/exports permet de déclarer les FS que l'on veut exporter en NFS.

Syntaxe NFS4 :

```
/exportnfs4       *(sec=sys,rw,fsid=0,insecure,no_subtree_check,async)
/exportnfs4/users *(sec=sys,rw,nohide,insecure,no_subtree_check,async)
```

Syntaxe NFS3 :

```
/exportnfs3       *(rw,sync,no_subtree_check)
```

Expliquez ces options.

Faire :

```
chmod 1777 /home/users
mount --bind /home/users /exportnfs4/users
mount -o remount,rw,bind /exportnfs4/users
```

Expliquez l'intérêt de cette commande. [3].

Démarrer le serveur :

```
systemctl start nfs-kernel-server
```

2.3 Installation du client NFS

1. systemd-resolved répond par le nom de la machine sans le domaine en multicast ce qui bloque l'authentification Kerberos sur ssh qui s'attend à un FQDN
2. ssh avec Kerberos vérifie la résolution inverse et son absence sera bloquante sauf en positionnant rdns=false dans /etc/krb5.conf mais bon c'est une sécurité de plus
3. voir cet excellent thread `https://unix.stackexchange.com/questions/198590/what-is-a-bind-mount`

```
apt-get install nfs-common nfs4-acl-tools nfswatch
```

2.3.1 Installation du daemon Idmap

Deux utilisateurs identiques sur deux machines différentes peuvent ne pas avoir le même uid. Idmap se sert du nom d'utilisateur (utilisateur@DOMAINE) pour mappez l'uid local qui correspond à cet utilisateur.

Pour l'activer modifiez le fichier /etc/idmapd.conf comme pour le serveur (même domaine).

```
NEED_IDMAPD=yes
```

Activer IDMAP dans /etc/default/nfs-common.

Sans authentification Kerberos (sec=sys) idmap n'est pas utilisé par défaut. L'activer avec

```
echo "N" > /sys/module/nfsd/parameters/nfs4_disable_idmapping
```

Pour rendre pérenne cette activation créez un fichier /etc/modprobe.d/nfs.conf contenant la ligne :

```
/etc/modprobe.d/nfs.conf
```

Mettez dans le fichier /etc/idmap.conf du client et du serveur :

```
Domain = iutbeziers.fr
```

2.3.2 Montages NFS en version 3 et 4

```
#Montez le partage en nfs4
mkdir /mnt/nfs4
mount -t nfs4 -o sec=sys nfsserveur.iutbeziers.fr:/users /mnt/nfs4
#En nfs3 :
mkdir /mnt/nfs3
mount -t nfs  nfsserveur.iutbeziers.fr:/exportnfs3 /mnt/nfs3
```

Quels sont les daemons qui sont nécessaires en NFS 3 et pas en NFS 4 ? Décrivez les daemon NFS4.

2.3.3 « Tips and tricks »

Commandes utiles :

```
showmount –e nom_du_serveur
exportfs  –a # permet de (re)publier les partages NFS
rpcinfo –p
/usr/bin/rpcinfo -u nfsserveur.iutbeziers.fr nfs
nfsstat –m
# permet de debuguer idmap (n'oubliez pas de killer ce daemon avant le relancer avant cette commande)
/usr/sbin/rpc.idmapd -f –vvvvvvvvvv
```

Vous pouvez modifier le fichier /etc/idmapd.conf pour augmenter le niveau de debug

```
[General]

Verbosity = 99
Pipefs-Directory = /run/rpc_pipefs
Domain = iutbeziers.fr
```

Lancez le daemon idmapd en mode debug :

```
/usr/sbin/rpc.idmapd -f –vvvvvvvvvv
```

3 Installation des outils de bench

Votre configuration de NFS 3 et 4 étant opérationnelle vous allez tester ses performances.

Vous vous servirez de la commande dd pour une première approximation : Il vous faut créer un fichier ayant deux fois la taille de la mémoire physique et démonter le partage entre chaque test.

Cette commande par exemple crée un fichier de 256 Mégas :

```
cd /mnt/nfs4
mkdir dirdd && cd dirdd
time dd if=/dev/zero of=./testfile bs=16k count=16384
```

3.1 Installation de netdata

Installez par package netdata. Il permettra d'avoir une vue globale de la machine mais aussi des performances NFS sur le client et le serveur NFS.

Il est accessible sur le port 19999 sur la "loopback" de la VM mais vous pouvez changer sa configuration dans /etc/netdata/netdata.conf.

3.2 Installation de filebench sur le client

Vous vous servirez aussi de l'utilitaire Sun Solaris Filebench porté sous Linux qui permet de faire des tests de charge de systèmes de fichiers dont NFS.

Chargez et compilez cet utilitaire. Voir `https://github.com/filebench/filebench`.

Installez les paquets suivants avant de compiler :

```
apt install libtool yacc bison flex haveged rng-tools
```

Utilisez varmail qui est un scenario "IO bound" simulant un serveur de messagerie sous NFS.

Avant de lancer le test saisir la commande suivante :

```
echo 0 | sudo tee /proc/sys/kernel/randomize_va_space
```

Copiez et adaptez le scenario /usr/local/share/filebench/workloads/varmail.f afin de tester vos filesystems locaux et nfs.

Vous devez tester, grapher et commentez.

l'article suivant :

`http://www.slashroot.in/how-do-linux-nfs-performance-tuning-and-optimization` peut vous inspirer sur les paramètres à faire varier. Pour être efficace décidez d'abord des paramètres (deux ou trois max) pour chaque test (FS local/nf4/nfs3/nfs4 krb5/nfs4 krb5i nfs4krb5p). Un graphique final comparatif est attendu.

4 Configuration de Kerberos

La configuration de Kerberos est délicate : faites un snapshot de votre machine virtuelle avant de lancer l'installation et la configuration afin de revenir facilement en arrière !

Pré-requis : Vous devez avoir défini les machines du royaume Kerberos (client et serveur DNS et SSH) au DNS Le service NTP doit être installé. Kerberos aime l'entropie pour générer de l'aléa dans les clefs de sécurité. Votre machine ou votre VM en a parfois trop peu. Installez les "rng-tools" et l'utilitaire haveged pour générer de l'entropie et vérifier son niveau. [4] L'entropie disponible sous votre système est visible via l'entrée suivante :

4. (voir `https://www.skyminds.net/serveur-dedie-produire-une-meilleure-reserve-dentropie-avec-haveged/` pour plus de détails)

```
cat /proc/sys/kernel/random/entropy_avail
```

Le test suivant doit montrer un nombre d'échecs inférieur à 5 :

```
cat /dev/random | rngtest -c 1000
```

Sinon configurez /etc/default/haveged avec DAEMON_ARGS="-w 2048" Si la réserve d'entropie est inférieure à ce seuil le daemon haveged se réveille pour en générer.

```
systemctl start haveged
```

Vous pouvez "débugguer" Kerberos via la variable d'environnement KRB5_TRACE :

```
export KRB5_TRACE=/dev/stdout
env KRB5_TRACE=/dev/stdout kadmin -p root/admin@IUTBEZIERS.FR
env KRB5_TRACE=/dev/stdout kvno p root/admin@IUTBEZIERS.FR
```

4.1 Installation du serveur Kerberos

Faites un snapshot de votre VM :

```
apt-get install krb5-admin-server krb5-kdc
```

Renommez le client et le serveur nfs avec le FQDN

```
hostnamectl set-hostname nfsserveur.iutbeziers.fr
hostnamectl set-hostname nfsclient.iutbeziers.fr
```

Donnez le nom du serveur comme KDC et comme serveur d'administration Kerberos.
Lancez cette commande pour initialiser le royaume :

```
krb5_newrealm
```

Le royaume correspond par convention au domaine DNS en MAJUSCULE de nfsserveur.iutbeziers.fr, soit IUTBEZIERS.FR
Modifiez :
/etc/krb5kdc/kadm5.acl
Pour autoriser la connexion au serveur d'administration Kerberos depuis la machine cliente.
Rajoutez les principals correspondant au client et au serveur à l'aide kadmin.local .

```
kadmin.local #
#ank -randkey host/fully qualified domain name_devotre machine
ank -randkey host/nfsclient.iutbeziers.fr
ank -randkey host/nfsserveur.iutbeziers.fr
```

A quoi sert l'option randkey ? Pourquoi est-elle nécessaire ? Rajouter le principal root/admin qui vous permettra de gérer votre Royaume. [5]

```
ank root/admin # avec son mot de passe
```

4.2 Installation du client Kerberos

.

```
apt-get install krb5-user
```

Configurez /etc/krb5.conf de votre client nfs.

5. Chantez la chanson de mel brooks "it's good to be the king ... oooh la la ..."

— Essayez sous l'utilisateur root de créer un ticket avec la commande kinit.
— Rajouter sur le serveur le compte nécessaire pour que ça fonctionne.
— Après avoir refait le kinit faite un klist.
— Pour quel service le ticket est-il obtenu ? Expliquez.
— Testez l'accès au serveur Kerberos via la commande kadmin.
— Testez la commande kdestroy

4.3 Installation d'un service ssh kerberisé

Le service ssh ne nécessite que la présence de principaux pour les "hôtes".
Installez un keytab du host pour que ça fonctionne via kadmin et ktadd. A quoi sert un keytab ? Sur le serveur SSH :

```
kadmin
ank -randkey nfs/nfsserver.iutbeziers.fr
ktadd host/nfsserveur.iutbeziers.fr@IUTBEZIERS.FR
```

Sur le client SSH :

```
ank -randkey nfs/nfsclient.iutbeziers.fr
ktadd host/nfsclient.iutbeziers.fr@IUTBEZIERS.FR
```

Vous devez autoriser dans le ficher /etc/ssh/sshd_config via l'option GSSAPIAuthentication.
Vérifiez que vous pouvez faire un ssh sans authentification après l'obtention d'un ticket Kerberos avec kinit entre serveur et client.

4.4 Kerbérisation de NFS

NFS nécessite en plus de principaux pour les "hosts" des principaux pour les services NFS. Créer les principaux nfs pour le serveur et le client
Pour authentifier NFS avec le service Kerberos , il va vous falloir créer deux principals pour le service client NFS et le service serveur NFS dans la base de données Kerberos avec l'option randkeys.
Ces deux principals sont exportés dans le krb5.keytab du client et du serveur pour nfs/serveur via la commande ktadd :

```
# sur le serveur
ktadd nfs/nfsserveur.iutbeziers.fr@IUTBEZIERS.FR
# sur le client
ktadd nfs/nfsclient.iutbeziers.fr@IUTBEZIERS.FR
```

Pour pouvoir fonctionner avec Kerberos , NFS4 va avoir besoin des service gss.
Activez-le aussi ans /etc/default/nfs-kernel-server et dans /etc/default/nfs-common pour le client.
Vous devez activez les partages suivants dans /etc/export :

```
/exportnfs4       *(sec=sys:krb5:krb5i:krb5p,rw,fsid=0,insecure,no_subtree_check,async)
/exportnfs4/users *(sec=sys:krb5:krb5i:krb5p,rw,nohide,insecure,no_subtree_check,async)
```

Monter l'export NFS en utilisant krb5 :

```
mount -t nfs4 -o sec=krb5 nfsserveur:/users /mnt/nfs4
```

Relancez filebench et faites varier l'option sec en utilisant krb5i et krb5p

4.4.1 Tips and tricks

En cas de difficultés vous pouvez lancez les daemons en mode debug :

```
/usr/sbin/rpc.svcgssd  -fvvvvvvv (serveur)
/usr/sbin/rpc.gssd  -fvvvvvvvrrrrrrrr (client)
```

Parfois il peut y avoir des divergences de version des clefs dans le fichier krb5.keytab.

```
# voir le contenu de krb5.keytab
klist -k
# vérifiez la version des clefs
kvno host/nfsserveur.iutbeziers.fr@IUTBEZIERS.FR
```

Si vous souhaitez créer les fichiers sur la partage nfs avec un user autre que nobody, modifiez /etc/idmap.conf pour mapper l'utilisateur (ici root) :

```
[General]

Verbosity = 99
Pipefs-Directory = /run/rpc\_pipefs
\# set your own domain here, if it differs from FQDN minus hostname
Domain = iutbeziers.fr

[Mapping]

Nobody-User = nobody
Nobody-Group = nogroup

[Translation]
Method=static,nsswitch
GSS-Methods = static,nsswitch

[Static]
nfs/nfsserveur.iutbeziers.fr@IUTBEZIERS.FR = root
nfs/nfsclient.iutbeziers.fr@IUTBEZIERS.FR = root
```