Manuel de Travaux Pratiques

Module « Administration Système en réseau »

Philippe Latu
philippe.latu(at)inetdoc.net

https://www.inetdoc.net

Résumé

Ce document présente la série de travaux pratiques du module sur l'administration système en réseau en première année de *Master mention Réseaux et télécommunication* de l'Université Paul Sabatier. Il se concentre sur deux aspects principaux : le stockage réseau et la gestion d'identité dans le contexte du « cloud privé » de la formation.

La première partie du document est consacrée aux technologies de stockage réseau : iSCSI et NFS.

iSCSI (*Internet Small Computer System Interface*) est un protocole de stockage en réseau qui caractérise les réseaux SAN (*Storage Area Network*). Il permet d'accéder à des unités de stockage distantes comme si elles étaient directement connectées au système local, en encapsulant des commandes SCSI dans des paquets IP. iSCSI établit une relation « 1 vers 1 » entre les rôles *target* (fournisseur de stockage) et *initiator* (consommateur de stockage).

NFS (*Network File System*), quant à lui, est caractéristique des réseaux NAS (*Network Attached Storage*). Il établit une relation « 1 vers n » entre un serveur NFS et plusieurs clients. NFS permet le partage de systèmes de fichiers sur un réseau, offrant aux clients un accès transparent aux fichiers et répertoires stockés sur le serveur comme s'ils étaient locaux.

La seconde partie du document se concentre sur la gestion d'identité à l'aide des annuaires LDAP (*Lightweight Directory Access Protocol*). Elle présente les principes de base des annuaires LDAP et guide les étudiants dans la configuration d'un serveur *OpenLDAP*. Les travaux pratiques incluent la création et la gestion d'un annuaire LDAP, la configuration de l'accès client, et l'intégration de LDAP avec d'autres services réseau tels que NFSv4 et *autofs* pour l'automontage des répertoires utilisateurs.

Toutes ces manipulations offrent une approche pratique et approfondie de l'administration système en réseau, combinant des technologies essentielles pour la gestion du stockage et des identités dans un environnement réseau moderne. Il permet aux étudiants d'acquérir des compétences concrètes et une compréhension approfondie des concepts clés de l'administration système distribuée.

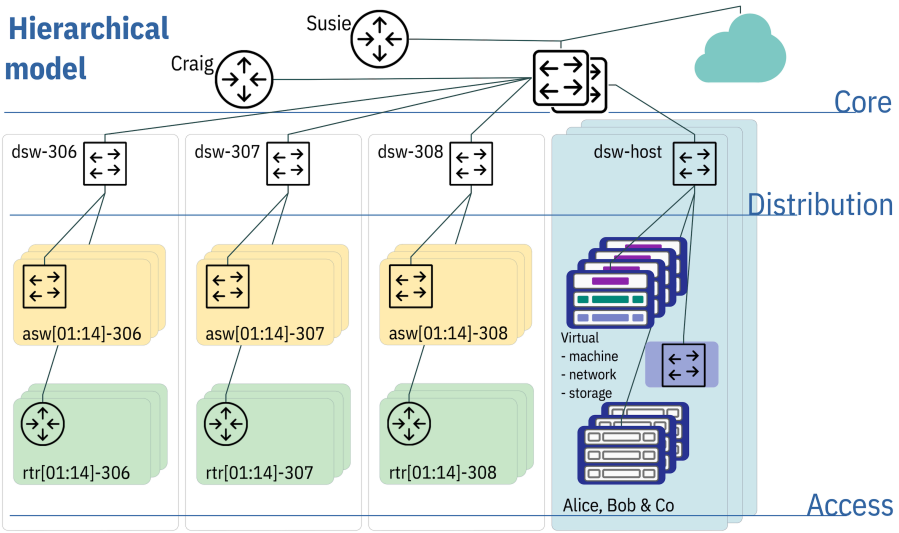


Table des matières

Introduction aux réseaux de stockage iSCSI

https://www.inetdoc.net

Résumé

Ce support de travaux pratiques est consacré à l'étude des technologies de stockage DAS (*Direct Attached Storage*), SAN(*Storage Area Network*) et de la redondance RAID1. Le protocole iSCSI est utilisé pour la partie SAN comme exemple d'accès «en mode bloc» aux unités de stockage réseau. La redondance RAID1 utilise les fonctions intégrées au noyau Linux. L'infrastructure proposée montre comment les différentes technologies élémentaires peuvent être combinées pour atteindre les objectifs de haute disponibilité et de sauvegarde.

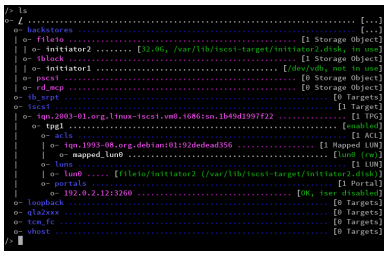


Table des matières

1. Objectifs

Après avoir réalisé les manipulations proposées par ce support, vous serez en mesure de :

Découvrir et comprendre les technologies de stockage réseau (DAS, SAN, iSCSI, RAID1).
Identifier les différences entre les architectures de stockage direct (DAS), les réseaux de stockage (SAN) et la redondance RAID1, en mettant l'accent sur le protocole iSCSI comme exemple d'accès en mode bloc aux unités de stockage réseau.

Configurer un initiateur iSCSI sur un système Linux.
Configurer une infrastructure minimale iSCSI, incluant la préparation des unités de stockage, la configuration des rôles initiator et target, et la validation de la connectivité réseau et du partage de volumes de stockage.

Expérimenter la redondance et la haute disponibilité avec RAID1.
Illustrer la création et la gestion d'un volume RAID1 combinant un disque local et un volume iSCSI, afin de démontrer la réplication synchrone des données et la tolérance aux pannes dans un environnement réel.

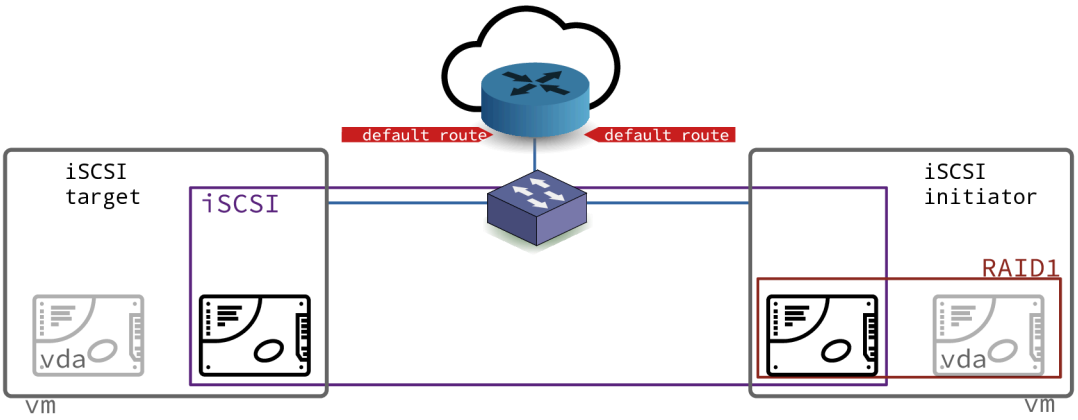
Automatiser la gestion et la sauvegarde des volumes logiques.
Initier les étudiants à la gestion avancée des volumes logiques (LVM), à la création de snapshots pour la sauvegarde, et à la restauration de données, tout en évaluant les performances des différentes solutions de stockage mises en œuvre.

2. Topologie, scénario et plan d'adressage

Topologie logique

Les manipulations présentées dans ce support utilisent un domaine de diffusion unique (VLAN) dans lequel on trouve deux systèmes virtuels ou physiques avec deux unités de stockage distinctes chacune.

- La première unité de stockage `/dev/vda` représente le stockage du système d'exploitation de la machine virtuelle.
- La deuxième unité de stockage `/dev/vdb` est dédiée aux manipulations présentées dans ce document.



Topologie logique - vue complète

Scénario

Le séquencement des opérations dépend des rôles définis par la technologie iSCSI.

Tableau 1. Attribution des rôles ISCSI

Rôle ***initiator***	Rôle ***target***
Préparation d'une unité de stockage locale en vue de la redondance avec l'unité de stockage réseau proposée par le rôle ***target***	Préparation d'une unité de stockage locale qui sera mise à disposition sur le réseau à l'aide de la technologie iSCSI
Recherche et installation du ou des paquet(s) pour le rôle ***initiator***	Recherche et installation du ou des paquet(s) pour le rôle ***target***
Étude des outils de configuration du service open-iscsi	Étude des outils de configuration du service targetcli
Validation manuelle de la configuration SAN iSCSI	
Validation de la configuration système	
Validation de l'authentification mutuelle entre les rôles ***initiator*** et ***target***	
Mise en place de la réplication synchrone avec un tableau RAID1 entre unité de disque locale et le volume iSCSI	Mise en place de la réplication asynchrone avec un volume logique de type ***snapshot*** de sauvegarde des fichiers images de volume de stockage
Étude comparative des performances d'accès	

Plan d'adressage

Partant de la topologie présentée ci-dessus, on utilise un plan d'adressage pour chacun des rôles iSCSI.

Le tableau ci-dessous correspond au plan d'adressage de la maquette qui a servi à traiter les questions des sections suivantes. Lors des séances de travaux pratiques, un plan d'adressage spécifique est fourni à chaque binôme d'étudiants. Il faut se référer au document *Infrastructure*.

Tableau 2. Plan d'adressage de la maquette « Introduction aux réseaux de stockage iSCSI »

Rôle	VLAN	Adresses IP
Initiator	369	`10.0.113.3/28`

Rôle	VLAN	Adresses IP
		`2001:678:3fc:171:baad:caff:fefe:6/64`
Target	369	`10.0.113.2/28` `2001:678:3fc:171:baad:caff:fefe:5/64`

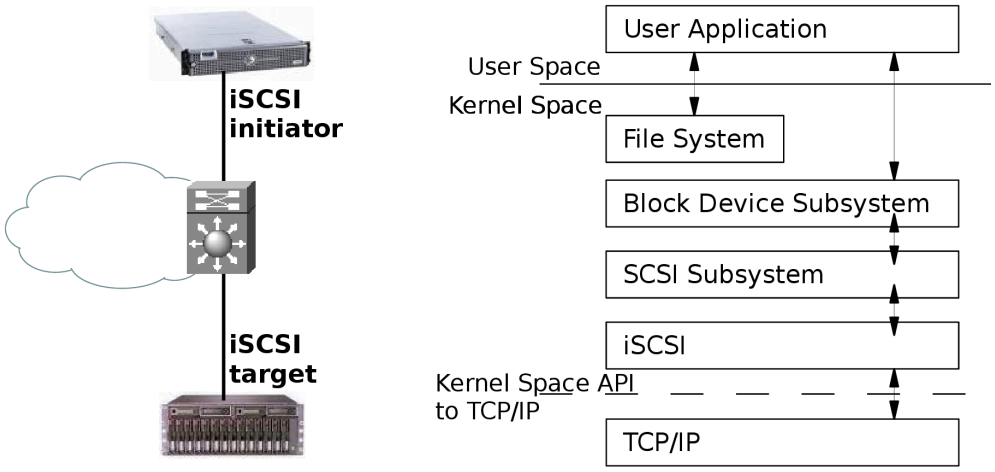
Pour traiter le scénario de ce support qui associe la technologie iSCSI, la redondance de disque RAID1 et la gestion de volume logique LVM, on utilise deux instances de machines virtuelle avec une unité de disque supplémentaire.

Avant de traiter les questions des sections suivantes, il faut rechercher dans le cours *Infrastructure* les éléments nécessaires au raccordement des machines virtuelles ou physiques. Les étapes usuelles sont les suivantes :

1. Attribuer les adresses IPv4 et IPv6 à chacun des postes en fonction de l'espace d'adressage du réseau défini.
2. Rechercher le numéro de VLAN correspondant aux réseaux IP attribués.
3. Repérer le commutateur sur lequel des ports ont été affectés au VLAN recherché. Connecter les deux postes de travaux pratiques sur les ports identifiés.
4. Configurer les interfaces réseau de chaque poste : adresse, masque et passerelle par défaut. Valider la connectivité IP entre les deux postes puis avec les autres réseaux de l'infrastructure de travaux pratiques.

3. Technologie iSCSI

Cette section présente sommairement le protocole iSCSI et les rôles de chacune des deux machines virtuelles ou physiques en fonction de la topologie mise en œuvre. Ce support fait suite à la présentation sur le *Stockage Réseau* utilisée en cours.



Topologie iSCSI basique - vue complète

La technologie iSCSI dont l'acronyme reprend la définition historique *Internet Small Computer System Interface* est un protocole réseau de stockage basé sur le modèle TCP/IP. Le principe de base consiste à encapsuler des commandes SCSI dans des paquets IP transmis entre un hôte et une unité de disque. Comme les paquets IP peuvent être perdus, retransmis ou ne pas arriver dans l'ordre d'émission. Le protocole iSCSI doit donc conserver une trace de la séquence de transmission de commandes SCSI. Les commandes sont placées dans une file d'attente dans l'ordre d'émission.

Le protocole iSCSI a initialement été développé par *IBM* et a ensuite été soumis à l'IETF (*Internet Engineering Task Force*). Le standard a été publié par le comité *IP Storage Working Group* en août 2002.

On peut identifier deux fonctions principales dans la technologie iSCSI. La première est la fonction *target*. C'est un système simple qui possède le volume de stockage à publier sur le réseau IP. Ce système peut être matériel

ou logiciel. Dans le cas de ces travaux pratiques, il s'agit d'un poste physique ou virtuel avec un second disque dur ou bien un fichier comme unité de stockage DAS. La seconde fonction est baptisée ***initiator***. Elle correspond au «client» qui utilise le volume de stockage réseau.

Fondamentalement, iSCSI est un protocole de la famille ***Storage Area Network*** (SAN). Le client ou ***initiator*** accède à une unité de stockage en ***mode bloc***. Ce mode de fonctionnement est quasi identique à la technologie ***Fibre Channel***. Le type de réseau constitue la principale différence entre ces deux technologies. La technologie iSCSI s'appuie sur TCP/IP alors que ***Fibre Channel*** comprend une définition de réseau propre (FC) qui nécessite des équipements spécifiques.

La technologie iSCSI a gagné en popularité relativement à son ainée pour plusieurs raisons.

- Le prix des configurations iSCSI peut être bien meilleur marché qu'avec la technologie ***Fibre Channel***. Si l'architecture du réseau de de stockage est adaptée, iSCSI devient très attractif.

  Il est important de bien identifier les fonctionnalités réseau que l'on associe à iSCSI pour accroître les performances du stockage. Dans ces fonctions complémentaires on trouve l'agrégation de canaux qui recouvre plusieurs dénominations et plusieurs standards de l'IEEE. Par exemple, elle est baptisée *bonding* sur les systèmes GNU/Linux et *etherchannel* sur les équipements ***Cisco***. Côté standard, le ***Link Aggregation Control Protocol*** (LACP) pour Ethernet est couvert par les versions ***IEEE 802.3ad***, ***IEEE 802.1aq*** et ***IEEE 802.1AX***. L'utilisation de ces techniques est totalement transparente entre équipements hétérogènes. Une autre technique consiste à utiliser aussi plusieurs liens dans une optique de redondance et de balance de charge. Elle est appelée *multipath*.

- L'utilisation d'une technologie réseau unique est nettement moins complexe à administrer. En effet, on optimise les coûts, les temps de formation et d'exploitation en utilisant une architecture de commutation homogène. C'est un des avantages majeurs de la technologie Ethernet sur ses concurrentes.

Aujourd'hui la technologie iSCSI est supportée par tous les systèmes d'exploitation communs. Côté GNU/Linux, plusieurs projets ont vu le jour dans les années qui ont suivi la publication du standard en 2002. Pour la partie ***initiator*** les développements des deux projets phares ont fusionné pour ne plus fournir qu'un seul code source ; celui disponible à l'adresse *Open-iSCSI*. La partie ***Kernelspace*** de ce dernier code est directement intégrée dans le noyau Linux. La mise en œuvre du rôle ***target*** ne nécessite donc que l'installation de la partie utilisateur pour paramétrer le sous-système de stockage du noyau.

```
$ aptitude search targetcli
p   targetcli-fb     - Command shell for managing the Linux LIO kernel target
```

Le choix du paquet pour le rôle ***initiator*** à l'aide de la liste ci-dessous est plus facile en combinant les deux critères de recherche. C'est le paquet open-iscsi qui convient.

```
$ aptitude search "?description(scsi)?description(initiator)"
p   open-iscsi            - iSCSI initiator tools
p   open-isns-discoveryd  - Internet Storage Name Service - iSNS discovery daemon
p   resource-agents       - Cluster Resource Agents
```

4. Préparer une unité de stockage

Dans cette section on présente les manipulations à effectuer pour préparer une unité de stockage à son utilisation dans une configuration DAS (et|ou) SAN.

> **Avertissement**
>
> Les copies d'écran utilisées dans les réponses correspondent à l'utilisation de machines virtuelles. Les unités de disques apparaissent donc sous le nom `/dev/vd[a-z]`. Les unités de disques physiques d'un système réel apparaissent sous le nom `/dev/sd[a-z]`.

4.1. Afficher la liste des unité de stockage

Pour commencer, il est utile de connaître la liste des unités de stockage en mode bloc sur un système.

Q1. Quelle est la commande apparentée à ls qui permet d'obtenir la liste des périphériques de stockage en mode bloc ?

Consulter la liste des outils forunis avec le paquet util-linux.

```
$ dpkg -L util-linux | grep bin/ls
/bin/lsblk
/usr/bin/lscpu
/usr/bin/lsipc
/usr/bin/lslocks
/usr/bin/lslogins
/usr/bin/lsmem
/usr/bin/lsns
```

Une fois que la commande lsblk est identifiée, on l'utilise pour obtenir la liste voulue.

```
$ lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
sr0     11:0    1  1024M  0 rom
vda    254:0    0   120G  0 disk
├─vda1 254:1    0   512M  0 part /boot/efi
├─vda2 254:2    0 118,5G  0 part /
└─vda3 254:3    0   977M  0 part [SWAP]
vdb    254:16   0    32G  0 disk
```

Dans le système de fichiers, c'est l'unité `/dev/vdb` qui doit être utilisée pour les manipulations de cette section.

4.2. Détruire la table des partitions

Sachant que les disques des postes de travaux pratiques physiques sont utilisés régulièrement, il est préférable de rendre l'unité de disque vierge de toute configuration.

Q2. Quelle est la syntaxe d'appel de l'outil parted qui permet de visualiser la table de partition d'une unité de disque ?

Consulter la documentation de parted à l'adresse *Using Parted*.

```
$ sudo parted /dev/vda print
Model: Virtio Block Device (virtblk)
Disk /dev/vda: 77,3GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Disk Flags:

Number  Start   End     Size    Type      File system     Flags
 1      1049kB  73,0GB  73,0GB  primary   ext4            boot
 2      73,0GB  77,3GB  4292MB  extended
 5      73,0GB  77,3GB  4292MB  logical   linux-swap(v1)
```

Q3. Quelle est la syntaxe de la commande dd qui permet d'effacer complètement la table des partitions d'une unité de disque ?

Utiliser l'aide en ligne de la commande : `dd --help`.

La commande suivante écrit des 0 dans les 4 premiers blocs de 512 octets de l'unité de disque.

```
$ sudo dd if=/dev/zero of=/dev/vdb bs=512 count=4
4+0 enregistrements lus
4+0 enregistrements écrits
2048 octets (2,0 kB, 2,0 KiB) copiés, 0,00621803 s, 329 kB/s

$ sudo parted /dev/vdb print
Error: /dev/vdb: unrecognised disk label
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 34,4GB
Sector size (logical/physical): 512B/512B
Partition Table: unknown
Disk Flags:
```

4.3. Créer une table des partitions et formater

Une fois que l'on dispose d'une unité de disque vierge, on peut passer à l'étape de création de la table des partitions. Cette opération n'est utile que pour traiter les questions de cette section.

La création de la table des partitions devra être reprise dans les deux contextes suivants :

- Le second disque du rôle ***initiator*** est destiné à intégrer l'unité logique RAID1. Il faudra donc créer une table de partition pour la nouvelle unité logique.
- Le disque réseau iSCSI est disponible une fois que la configuration du rôle ***target*** est active. Une fois la session iSCSI établie, l'unité logique réseau est la propriété exclusive du rôle ***initiator***.

Q4. Comment créer une partition unique couvrant la totalité de l'espace de stockage de l'unité de disque ?

Consulter la documentation de parted à l'adresse *Using Parted*.

```
$ sudo parted /dev/vdb
GNU Parted 3.4
Using /dev/vdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) print
Error: /dev/vdb: unrecognised disk label
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 34,4GB
Sector size (logical/physical): 512B/512B
Partition Table: unknown
Disk Flags:
(parted) mklabel gpt
(parted) print
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 34,4GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start  End  Size  File system  Name  Flags

(parted) mkpart ext4 0% 100%
(parted) print
Model: Virtio Block Device (virtblk)
Disk /dev/vdb: 34,4GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name  Flags
 1      1049kB  34,4GB  34,4GB               ext4

(parted) quit
Information: You may need to update /etc/fstab.
```

Q5. Quelle est la commande à utiliser pour les opérations de formatage ? Quel est le rôle de l'option `-T` de cette commande ?

Les informations utiles sont disponibles à la page *Ext4 Howto*. Les pages de manuels détaillent les fonctions des options.

La commande utilisée pour le formatage d'un système de fichiers `ext4`.

```
$ dpkg -S `which mkfs.ext4`
e2fsprogs: /sbin/mkfs.ext4
```

L'option `-T` définit le type d'utilisation du système de fichiers à formater suivant sa taille. Les paramètres par défaut sont les suivants :

- `floppy` : 0 < taille < 3Mo
- `small` : 3Mo < taille < 512Mo
- `default` : 512Mo < taille < 4To
- `big` : 4To < taille < 16To
- `huge` : 16To < taille

Q6. Quelle est la syntaxe de la commande de formatage de la partition créée lors de l'étape précédente ?

Des exemples de syntaxe sont disponibles à la page *Ext4 Howto*.

```
$ sudo mkfs.ext4 /dev/vdb1
mke2fs 1.46.2 (28-Feb-2021)
Rejet des blocs de périphérique : complété
En train de créer un système de fichiers avec 8388096 4k blocs et 2097152 i-noeuds.
UUID de système de fichiers=7c582ccd-ce99-43ec-b145-05f043c02fc6
Superblocs de secours stockés sur les blocs :
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624

Allocation des tables de groupe : complété
Écriture des tables d'i-noeuds : complété
Création du journal (32768 blocs) : complété
Écriture des superblocs et de l'information de comptabilité du système de
fichiers : complété
```

Q7. Quelle est la syntaxe de la commande de visualisation des attributs du système de fichiers créé lors du formatage ?

Les informations utiles sur les attributs sont fournies à la page *Ext4 Howto*.

```
$ sudo tune2fs -l /dev/vdb1
tune2fs 1.46.2 (28-Feb-2021)
Filesystem volume name:   <none>
Last mounted on:          <not available>
Filesystem UUID:          7c582ccd-ce99-43ec-b145-05f043c02fc6
Filesystem magic number:  0xEF53
Filesystem revision #:    1 (dynamic)
Filesystem features:      has_journal ext_attr resize_inode dir_index filetype extent 64bit flex_bg sparse_super large_file
Filesystem flags:         signed_directory_hash
Default mount options:    user_xattr acl
Filesystem state:         clean
Errors behavior:          Continue
Filesystem OS type:       Linux
Inode count:              2097152
Block count:              8388096
Reserved block count:     419404
Overhead clusters:        176700
Free blocks:              8211390
Free inodes:              2097141
First block:              0
Block size:               4096
Fragment size:            4096
Group descriptor size:    64
Reserved GDT blocks:      1024
Blocks per group:         32768
Fragments per group:      32768
Inodes per group:         8192
Inode blocks per group:   512
Flex block group size:    16
Filesystem created:       Sat Aug 21 17:14:07 2021
Last mount time:          n/a
Last write time:          Sat Aug 21 17:14:07 2021
Mount count:              0
Maximum mount count:      -1
Last checked:             Sat Aug 21 17:14:07 2021
Check interval:           0 (<none>)
Lifetime writes:          4182 kB
Reserved blocks uid:      0 (user root)
Reserved blocks gid:      0 (group root)
First inode:              11
Inode size:               256
Required extra isize:     32
Desired extra isize:      32
Journal inode:            8
Default directory hash:   half_md4
Directory Hash Seed:      df4bc602-36c1-4a6c-8bd0-cc7bc6809114
Journal backup:           inode blocks
Checksum type:            crc32c
Checksum:                 0xa952ed62
```

4.4. Monter manuellement un volume de stockage

Une fois qu'un volume de stockage a été partitionné et formaté, on peut le ***"monter"*** dans l'arborescence du système de fichiers du système de façon à pouvoir lire et écrire des données.

Q8. Comment obtenir l'identifiant du volume de stockage à ajouter au système de fichiers ?

Consulter la liste des utilitaires fournis avec le paquet util-linux. Il faut se rappeler que la représentation fichier d'un périphérique de stockage se distingue par son mode d'accès : le mode bloc.

La commande à utiliser est blkid. Dans l'exemple de la partition `/dev/vdb1`, on obtient le résultat suivant.

```
$ sudo blkid /dev/vdb1
/dev/vdb1: UUID="7c582ccd-ce99-43ec-b145-05f043c02fc6" BLOCK_SIZE="4096" TYPE="ext4" \
  PARTLABEL="ext4" PARTUUID="244bacd9-38ca-44e4-8ab7-16d5f2c85f98"
```

Q9. Dans quel fichier de configuration trouve-t-on la liste des périphériques montés lors de l'initialisation du système ?

Consulter la liste des fichiers du paquet util-linux.

Le fichier recherché est `/etc/fstab`. Il contient la liste des points de montage. Dans l'exemple ci-dessous, la racine et la partition d'échange utilisée en cas de saturation des ressources RAM du système.

```
$ grep -v '^#' /etc/fstab
UUID=8362b3e6-d426-4f1b-93eb-e1efc22f60f4 /       ext4    errors=remount-ro 0       1
UUID=f3e18b95-7430-4fea-ace5-7dd4cea6398a none    swap    sw              0       0
/dev/sr0        /media/cdrom0   udf,iso9660 user,noauto     0       0
```

Q10. Quelle est la commande qui donne la liste des montages en cours d'utilisation sur le système ? Quelle est l'option qui permet de scruter les entrées du fichier recherché dans la question précédente et de monter tous les points non encore utilisés ?

La commande est fournie par le paquet du même nom.

Le paquet mount fournit la commande du même nom. Cette commande liste tous les montages actifs du système. La liste comprend les systèmes de fichiers virtuels qui représentent l'état courant des paramètres du noyau ainsi que les systèmes de fichiers physiques qui correspondent aux volumes de stockage effectifs. En reprenant l'exemple utilisé auparavant et en filtrant les systèmes de fichiers virtuels, on obtient :

```
$ mount | grep "/dev/vd"
/dev/vda1 on / type ext4 (rw,relatime,errors=remount-ro)
```

L'option de montage des entrées inutilisées du fichier `/etc/fstab` est `-a`. Elle doit être utilisée dans la question suivante.

Q11. Comment monter manuellement le système de fichiers de la partition `/dev/vdb1` ?

Le répertoire de test pour les montages temporaires est historiquement `/mnt/`.

Consulter les pages de manuels de la commande mount.

On dispose d'au moins deux solutions pour désigner la partition à monter.

- Utilser l'identifiant de partition unique.

```
$ sudo mount -U 7c582ccd-ce99-43ec-b145-05f043c02fc6 /mnt
```

- Utiliser le nom défini par udev dans le système de fichiers.

```
$ sudo mount /dev/vdb1 /mnt
```

Pour terminer, on liste les montages pour vérifier que la nouvelle partition est bien présente.

```
$ mount | grep "/dev/vd"
/dev/vda1 on / type ext4 (rw,relatime,errors=remount-ro)
/dev/vdb1 on /mnt type ext4 (rw,relatime)
```

Q12. Comment démonter manuellement le système de fichiers de la partition `/dev/vdb1` ?

Consulter les pages de manuels de la commande mount.

L'opération de démontage utilise l'arborescence du système de fichiers pour désigner le volume de stockage.

```
$ sudo umount /mnt
$ mount | grep "/dev/vd"
/dev/vda1 on / type ext4 (rw,relatime,errors=remount-ro)
```

5. Configurer le système initiator

Dans cette partie, on prépare le système auquel on a attribué le rôle ***initiator***. Ce système est celui qui utilise le volume de stockage mis à disposition sur le réseau par le rôle ***target***.

5.1. Sélectionner le paquet et lancer le service

Q13. Comment identifier et installer le paquet correspondant au rôle ***initiator*** ?

En effectuant une recherche simple dans le catalogue des paquets disponibles, on obtient la liste des paquets dont le nom contient la chaîne de caractères `iscsi`.

```
$ aptitude search iscsi
p   iscsitarget         - iSCSI Enterprise Target userland tools
p   iscsitarget-dkms    - iSCSI Enterprise Target kernel module source - dkms version
p   iscsitarget-source  - iSCSI Enterprise Target kernel module source
p   open-iscsi          - High performance, transport independent iSCSI implementation
```

On remarque que le paquet `open-iscsi` est le seul qui ne soit pas identifié comme appartenant à la catégorie ***target***.

```
$ sudo apt install open-iscsi
```

Q14. Comment connaître l'état du service ***initiator*** et valider son fonctionnement ?

À partir de la liste des services actifs, on repère les message relatifs au rôle ***initiator***.

```
$ systemctl status open-iscsi.service
# open-iscsi.service - Login to default iSCSI targets
     Loaded: loaded (/usr/lib/systemd/system/open-iscsi.service; enabled; preset: enabled)
     Active: inactive (dead)
  Condition: start condition unmet at Fri 2024-08-30 10:02:44 CEST; 7s ago
             ├─ ConditionDirectoryNotEmpty=|/etc/iscsi/nodes was not met
             └─ ConditionDirectoryNotEmpty=|/sys/class/iscsi_session was not met
       Docs: man:iscsiadm(8)
             man:iscsid(8)

août 30 09:57:40 initiator systemd[1]: open-iscsi.service - Login to default iSCSI targets was skipped because no trigger c
août 30 10:02:44 initiator systemd[1]: open-iscsi.service - Login to default iSCSI targets was skipped because no trigger c
```

Le lancement du service se fait de façon classique avec systemd.

```
$ sudo systemctl restart open-iscsi
```

Avertissement

L'état actuel de la configuration montre que le service est lancé sans aucune session iSCSI active. Pour l'instant aucun système avec le rôle ***target*** n'a été contacté.

5.2. Accéder aux volumes de stockage réseau iSCSI

Q15. Quelle est la commande principale du rôle ***initiator*** qui permet de tester la connectivité iSCSI ?

Consulter la liste des fichiers du paquet `open-iscsi`.

En consultant la liste donnée ci-dessus, on ne relève qu'un seul outil exécutable : la commande iscsiadm.

Q16. Quelles sont les options de découverte proposées avec cette commande ? Donner un exemple fournissant l'identifiant de l'unité de stockage réseau visible.

Consulter les pages de manuels de la commande identifiée dans la question précédente.

À partir du système ***initator***, on liste le ou les volume(s) de stockage visible sur le réseau local :

Si le portail du système avec le rôle ***target*** est configuré pour être accessible via IPv6, on peut utiliser la commande suivante en adaptant l'adresse au contexte :

```
$ sudo iscsiadm -m discovery \
 --type sendtargets \
 --portal=[2001:678:3fc:171:baad:caff:fefe:5]
[2001:678:3fc:171:baad:caff:fefe:5]:3260,1 iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660
```

```
$ sudo iscsiadm -m discovery \
 --type sendtargets \
 --portal=10.0.20.131
10.0.20.131:3260,1 iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660
```

Dans les deux copies d'écran ci-dessus, l'identifiant du volume de stockage réseau visible est `iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660`.

Malheureusement, les adresses de lien local IPv6 ne sont pas utilisables au moment de la rédaction de ces lignes.

Q17. Comment obtenir la liste des portails iSCSI déjà connus du système ***initiator*** ?

Rechercher dans les pages de manuels de la commande iscsiadm.

C'est le mode `node` qui permet d'obtenir l'information demandée.

```
$ sudo iscsiadm -m node
[2001:678:3fc:171:baad:caff:fefe:5]:3260,1 iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660
```

Q18. Comment effacer la liste des portails iSCSI déjà connus du système ***initiator*** ?

Rechercher dans les pages de manuels de la commande iscsiadm.

C'est le mode `node` qui permet d'obtenir l'information demandée.

```
$ sudo iscsiadm -m node --op=delete
```

Avertissement

Attention ! Si la commande ci-dessus est exécutée, il faut reprendre les opérations de découverte décrites à la question Q : Q16 pour compléter la liste des portails iSCSI connus.

Q19. Quel est l'identifiant à communiquer ou à paramétrer pour que le système *initiator* soit reconnu côté système *target* ?

Rechercher les informations relatives au nommage iSCSI dans les outils et les fichiers fournis avec le paquet de gestion du rôle *initiator*.

Le répertoire `/etc/iscsi/` contient les paramètres de configuration du service.

```
$ ls -p /etc/iscsi/
initiatorname.iscsi  iscsid.conf  nodes/  send_targets/
```

On consulte ou on édite ce fichier de façon à communiquer l'identité du système *initiator* au système *target* pour configurer le contrôle d'accès.

Par exemple, l'identifiant unique donnée dans la copie d'écran ci-dessous est à transmettre au système *target*.

```
$ sudo grep -v ^# /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.1993-08.org.debian:01:2cc8dac75cec
```

Côté *target*, on obtient le résultat suivant après avoir créé la liste de contrôle d'accès au volume réseau via l'interface targetcli.

```
sudo targetcli
targetcli shell version 2.1.53
Copyright 2011-2013 by Datera, Inc and others.
For help on commands, type 'help'.

/> ls
o- / ......................................................................... [...]
  o- backstores .............................................................. [...]
  | o- block .................................................. [Storage Objects: 1]
  | | o- blockvol0 ....................../dev/vdb (32.0GiB) write-thru activated]
  | |   o- alua ....................................................... [ALUA Groups: 1]
  | |     o- default_tg_pt_gp ..................... [ALUA state: Active/optimized]
  | o- fileio ................................................. [Storage Objects: 1]
  | | o- filevol0 .......................levol0 (32.0GiB) write-back deactivated]
  | |   o- alua ....................................................... [ALUA Groups: 1]
  | |     o- default_tg_pt_gp ..................... [ALUA state: Active/optimized]
  | o- pscsi .................................................. [Storage Objects: 0]
  | o- ramdisk ................................................ [Storage Objects: 0]
  o- iscsi ............................................................ [Targets: 1]
  | o- iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660 ..... [TPGs: 1]
  |   o- tpg1 ............................................... [no-gen-acls, no-auth]
  |     o- acls ......................................................... [ACLs: 1]
  |     | o- iqn.1993-08.org.debian:01:2cc8dac75cec ............. [Mapped LUNs: 1]
  |     |   o- mapped_lun0 ........................... [lun0 block/blockvol0 (rw)]
  |     o- luns ......................................................... [LUNs: 1]
  |     | o- lun0 ................ [block/blockvol0 (/dev/vdb) (default_tg_pt_gp)]
  |     o- portals ................................................... [Portals: 1]
  |       o- [::0]:3260 ..................................................... [OK]
  o- loopback ......................................................... [Targets: 0]
  o- vhost ............................................................ [Targets: 0]
  o- xen-pvscsi ....................................................... [Targets: 0]
```

La copie d'écran ci-dessus montre l'association des identités iSCSI des systèmes *initiator* et *target*.

Q20. Quelles sont les options de connexion proposées avec cette même commande ?

Donner un exemple illustrant l'établissement d'une connexion.

Consulter les pages de manuels de la commande identifiée précédemment.

```
$ sudo iscsiadm -m node \
 -T iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660 \
 -p [2001:678:3fc:171:baad:caff:fefe:5] \
 -l
Logging in to [iface: default,
  target: iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660,
  portal: 2001:678:3fc:171:baad:caff:fefe:5,3260]
Login to [iface: default,
  target: iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660,
  portal: 2001:678:3fc:171:baad:caff:fefe:5,3260] successful.
```

Dans l'exemple ci-dessus, la connexion sans authentification est un succès dans la mesure où les paramètres d'authentification et de protection en écriture ont été forcés à zéro sur la configuration du système *target*. Voir la section intitulée « Partie portail iSCSI »

Q21. Comment obtenir les caractéristiques de l'unité de stockage iSCSI associée ?

Revoir la question Quelle est la commande apparentée à ls qui permet d'obtenir la liste des périphériques de stockage en mode bloc ? et/ou consulter les journaux système.

Le résultat de la commande lsblk montre l'arrivée d'un nouveau volume de stockage.

```
$ sudo lsblk
NAME   MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda      8:0    0   32G  0 disk
└─sda1   8:1    0   32G  0 part
sr0     11:0    1 1024M  0 rom
vda    254:0    0   72G  0 disk
├─vda1 254:1    0   68G  0 part /
├─vda2 254:2    0    1K  0 part
└─vda5 254:5    0    4G  0 part [SWAP]
vdb    254:16   0   32G  0 disk
```

Voici un extrait des messages de journalisation du système.

```
$ journalctl -n 20 -f --grep '(sd|scsi)'
initiator kernel: scsi host7: iSCSI Initiator over TCP/IP
initiator kernel: scsi 7:0:0:0: Direct-Access     LIO-ORG  sda               4.0  PQ: 0 ANSI: 6
initiator kernel: sd 7:0:0:0: Attached scsi generic sg1 type 0
initiator kernel: sd 7:0:0:0: [sdb] 67108864 512-byte logical blocks: (34.4 GB/32.0 GiB)
initiator kernel: sd 7:0:0:0: [sdb] Write Protect is off
initiator kernel: sd 7:0:0:0: [sdb] Mode Sense: 43 00 10 08
initiator kernel: sd 7:0:0:0: [sdb] Write cache: enabled, read cache: enabled, supports DPO and FUA
initiator kernel: sd 7:0:0:0: [sdb] Preferred minimum I/O size 512 bytes
initiator kernel: sd 7:0:0:0: [sdb] Optimal transfer size 33550336 bytes
initiator kernel: sd 7:0:0:0: [sdb] Attached SCSI disk
initiator iscsid[620]: Connection1:0 to [target: iqn.2003-01.org.linux-iscsi.target.x8664:sn.368ea864a0dd, portal: 2001:678
```

Q22. Donner la liste des entrées de périphériques de stockage créées par le démon `udev` ?

Lister les entrées de périphériques mode bloc de l'arborescence système.

Les fichiers de description des périphériques mode bloc sont tous situés dans le répertoire `/dev/`. En reprenant l'exemple ci-dessus, on obtient :

```
$ ls -lA /dev/[v,s]d*
brw-rw---- 1 root disk   8,  0 22 août  19:17 /dev/sda
brw-rw---- 1 root disk   8,  1 22 août  19:17 /dev/sda1
brw-rw---- 1 root disk 254,  0 22 août  19:13 /dev/vda
brw-rw---- 1 root disk 254,  1 22 août  19:13 /dev/vda1
brw-rw---- 1 root disk 254,  2 22 août  19:13 /dev/vda2
brw-rw---- 1 root disk 254,  5 22 août  19:13 /dev/vda5
brw-rw---- 1 root disk 254, 16 22 août  19:13 /dev/vdb
```

L'entrée `/dev/sda` correspond à l'unité de disque iSCSI. Le volume de stockage est donc bien vu de façon transparente comme un périphérique local du système accessible en mode bloc. Il entre bien dans la catégorie SAN ou *Storage Area Network*.

5.3. Réinitialiser la session iSCSI

Dans le cas d'une reconfiguration avec un autre hôte *target* ou dans le cas d'un dépannage, il est utile de pouvoir reprendre les paramètres du rôle *initiator*.

Q23. Comment obtenir la liste des sessions actives avec le système *target* ?

Consulter les pages de manuels de la commande de configuration du rôle *initiator* : iscsiadm.

C'est le mode `session`, documenté dans les pages de manuels de la commande iscsiadm, qui permet de répondre à la question.

```
$ sudo iscsiadm -m session
tcp: [2] [2001:678:3fc:171:baad:caff:fefe:5]:3260,1
 iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660 (non-flash)
```

Q24. Comment libérer toutes les sessions actives depuis le système *initiator* ?

Consulter les pages de manuels de la commande de configuration du rôle *initiator* : iscsiadm.

Pour cette question, c'est le mode `node` qui nous intéresse.

```
$ sudo iscsiadm -m node -U all
Logging out of session [sid: 2, target:
 iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660,
 portal: 2001:678:3fc:171:baad:caff:fefe:5,3260]
Logout of [sid: 2, target:
 iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660,
 portal: 2001:678:3fc:171:baad:caff:fefe:5,3260] successful.
```

Bien sûr, il faut relancer une nouvelle session iSCSI pour traiter les manipulations suivantes.

5.4. Configuration système permanente

Une fois la connexion à la ressource iSCSI testée, on peut passer à la configuration système de façon à retrouver le volume de stockage après une réinitialisation du système *initiator*.

Q25. Comment rendre la connexion à l'unité de stockage automatique lors de l'initialisation du système *initiator* ?

Rechercher dans la liste des fichiers du paquet open-iscsi les éléments relatifs à la configuration système. Éditer le fichier de configuration principal de façon à rendre automatique le lancement du service.

Au niveau système, les fichiers de configuration sont nécessairement dans le répertoire `/etc/`.

```
$ dpkg -L open-iscsi | grep '/etc/'
/etc/default
/etc/default/open-iscsi
/etc/init.d
/etc/init.d/iscsid
/etc/init.d/open-iscsi
/etc/iscsi
/etc/iscsi/iscsid.conf
```

Le fichier `/etc/iscsi/iscsid.conf` contient une directive dans la section ***Startup settings*** qui rend automatique l'accès à une ressource déjà enregistrée. Voici le contenu de cette section extraite du fichier de configuration.

```
#*****************
# Startup settings
#*****************

# To request that the iscsi initd scripts startup a session set to "automatic".
node.startup = automatic
```

Avertissement

Attention ! Après édition du fichier `/etc/iscsi/iscsid.conf`, la valeur `automatic` n'est appliquée que pour les nouvelles opérations de découverte et d'ouverture de session.

Pour rendre ce l'ouverture de session automatique au démarrage du système, il faut clore les sessions en cours et effacer les informations de découverte.

Voici un exemple qui donne la séquence des opérations.

```
$ sudo iscsiadm -m node -U all
$ sudo iscsiadm -m node --op=delete
$ sudo iscsiadm -m discovery \
 --type sendtargets \
 --portal=[2001:678:3fc:171:baad:caff:fefe:5]
$ sudo iscsiadm -m node \
 -T iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660 \
 -p [2001:678:3fc:171:baad:caff:fefe:5] \
 -l
$ sudo grep \\.startup /etc/iscsi/nodes/iqn.2003-01.org.linux-iscsi.target-vm.x8664\:sn.bc4899490660/2001\:678\:3fc\:171\:b
node.startup = automatic
node.conn[0].startup = manual
```

Q26. Comment connaître l'état et la liste d'une session iSCSI active ?

Consulter les pages de manuels de la commande de configuration du rôle ***initiator*** : iscsiadm.

Il existe un mode `session` dédié aux manipulations sur les sessions. La commande de test la plus simple est la suivante.

```
$ sudo iscsiadm -m session
tcp: [1] [2001:678:3fc:171:baad:caff:fefe:5]:3260,1 \
 iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660 (non-flash)
```

Si la liste est vide, il n'y a pas de session iSCSI active en cours.

Il est possible d'obtenir davantage d'informations sur les paramètres de session en cours à l'aide de l'option `-P` suivie d'un numéro désignant le niveau de détail attendu.

La commande `iscsiadm -m session -P 3` affiche les paramètres sur les interfaces réseau utilisées, etc.

Q27. Comment retrouver un point de montage unique du volume de stockage iSCSI après réinitialisation du système *initiator* ?

Créer un répertoire de montage et rechercher les options utiles dans les pages de manuels des commandes mount, systemd.mount et blkid. Éditer le fichier `/etc/fstab` en utilisant les options sélectionnées. Noter que le fichier `fstab` possède ses propres pages de manuels.

La création du répertoire destiné au montage du volume de stockage iSCSI ne pose pas de problème.

```
$ sudo mkdir /var/cache/iscsi-vol0
```

C'est à cette étape que les question de la Section 4, « Préparer une unité de stockage » sont utiles.

Après partitionnement de l'unité de stockage iSCSI `/dev/sda` et formatage de la partition `/dev/sda1`, on peut relever l'identifiant unique de ce volume avec la commande blkid. Voici un exemple.

```
$ sudo lsblk /dev/sda1
NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINTS
sda1   8:1    0  32G  0 part
$ sudo blkid /dev/sda1
/dev/sda1: UUID="4df99b8b-0021-44bd-b751-bd180f018200"
 UUID_SUB="0f2453f9-61b5-49d2-93ff-4aafd3ca0969"
 BLOCK_SIZE="4096"
 TYPE="btrfs"
 PARTLABEL="vol0"
 PARTUUID="fb154fb0-afc4-4a89-8e67-44b9d5fa8a05"
```

Q28. Quelles sont les informations à insérer dans le fichier `/etc/fstab` pour assurer le montage du volume de stockage à chaque initialisation du système ?

Consulter les pages de manuels de la commande mount ainsi que la documentation du paquet open-iscsi.

Le choix des options à utiliser lors de l'édition du fichier `/etc/fstab` constitue un point très délicat.

```
echo "UUID=4df99b8b-0021-44bd-b751-bd180f018200 \
 /var/cache/iscsi-vol0 \
 btrfs \
 _netdev \
 0   2" | sudo tee -a /etc/fstab
```

- Le choix de la valeur `UUID` se fait à partir du résultat de la commande blkid donné ci-dessus.
- Le point de montage `/var/cache/iscsi-vol0` a lui aussi été défini ci-dessus.
- Le système de fichiers utilisé est, là encore, connu : `btrfs`.
- L'option `_netdev` spécifie que le système de fichiers réside sur un périphérique nécessitant des accès réseau. Il est donc inutile d'y accéder tant qu'aucune interface réseau n'est active.

6. Configuration du système target

Dans cette partie, on prépare le système auquel on a attribué le rôle ***target*** à l'aide de l'outil targetcli-fb.

6.1. Installation de l'outil de paramétrage du rôle target

Q29. Quel est le paquet qui contient l'outil de configuration du service dans l'espace utilisateur ?

On recherche le mot clé ***`targetcli`*** dans la liste des paquets.

```
$ apt search ^targetcli
En train de trier... Fait
Recherche en texte intégral... Fait
targetcli-fb/testing,now 1:2.1.53-1 all
  Command shell for managing the Linux LIO kernel target
```

Q30. Comment installer le paquet identifié à la question précédente ?

```
$ sudo apt install targetcli-fb
```

6.2. Configuration du rôle target

La technologie iSCSI dispose d'un schéma de nommage propre défini dans le document standard RFC3721 *Internet Small Computer Systems Interface (iSCSI) Naming and Discovery*. Le format retenu ici est baptisé iqn (*iSCSI Qualified Name*). Il s'agit d'une chaîne qui débute par `iqn.` suivie d'une date au format `AAAA-MM`, du nom de l'autorité qui a attribué le nom (le nom de domaine à l'envers), puis une autre chaîne unique qui identifie le nœud de stockage.

Dans un premier temps, on n'utilise aucun mécanisme d'authentification sachant que la configuration initiale se fait dans un contexte de travaux pratiques sur un réseau isolé.

Q31. Quelles sont les étapes à suivre pour publier un volume de stockage sur le réseau à partir de l'interface de l'outil targetcli ?

On commence par identifier les deux entrées intéressantes à partir du menu prinicpal de l'outil de configuration targetcli.

```
$ sudo targetcli
targetcli shell version 2.1.53
Copyright 2011-2013 by Datera, Inc and others.
For help on commands, type 'help'.

/> ls
o- / ......................................................... [...]
  o- backstores .............................................. [...]
  | o- block ...................................... [Storage Objects: 0]
  | o- fileio ..................................... [Storage Objects: 0]
  | o- pscsi ...................................... [Storage Objects: 0]
  | o- ramdisk .................................... [Storage Objects: 0]
  o- iscsi ................................................ [Targets: 0]
  o- loopback ............................................. [Targets: 0]
  o- vhost ................................................ [Targets: 0]
  o- xen-pvscsi ........................................... [Targets: 0]
/>
```

- La section ***backstores*** désigne les volumes de stockage à publier sur le réseau. Ici, les deux items intéressants sont ***fileio*** et ***block***. Le premier fait correspondre un fichier du système local au volume à publier. Le second fait correspondre une unité de disque physique au volume à publier.
- La section ***iscsi*** sert à définir une «cible» (*target*) qui comprend au moins une unité logique (LUN en vocabulaire SCSI). C'est ici que l'on configure le point de contact réseau pour le système *initiator*.

Partie stockage local : *backstores*

Q32. Quelles sont les opérations à effectuer définir un disque physique comme volume de stockage ?

Consulter le site de référence et repérer les options du menu `block`.

On créé un volume appelé ***blockvol0*** associé à l'unité de stockage locale au système `/dev/vdb`.

```
/> cd /backstores/block
/backstores/block> create blockvol0 /dev/vdb
Created block storage object blockvol0 using /dev/vdb.
/backstores/block> ls
o- block ........................................... [Storage Objects: 1]
  o- blockvol0 ............. [/dev/vdb (32.0GiB) write-thru deactivated]
    o- alua ............................................ [ALUA Groups: 1]
      o- default_tg_pt_gp ............... [ALUA state: Active/optimized]
```

Q33. Quelles sont les opérations à effectuer pour définir un fichier comme volume de stockage ?

Consulter le site de référence et repérer les options du menu `fileio`.

On créé un volume appelé ***filevol0*** associé au fichier `/var/cache/filevol0`.

```
/> cd /backstores/fileio
/backstores/fileio> create filevol0 /var/cache/filevol0 32G
Created fileio filevol0 with size 34359738368
/backstores/fileio> ls
o- fileio ......................................... [Storage Objects: 1]
  o- filevol0 ... [/var/cache/filevol0 (32.0GiB) write-back deactivated]
    o- alua ............................................ [ALUA Groups: 1]
      o- default_tg_pt_gp ............... [ALUA state: Active/optimized]
```

Partie portail iSCSI

Q34. Quelles sont les opérations à effectuer pour définir un nouveau portail réseau iSCSI ?

Consulter le site de référence et repérer les options du menu `iscsi`. Attention ! Une cible iSCSI comprend plusieurs attributs.

1. Nommage du portal au format iqn.
   Si le nom du portail n'est pas fourni avec la commande create, il est généré automatiquement.

```
/> cd /iscsi
/iscsi> create
Created target iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660.
Created TPG 1.
Global pref auto_add_default_portal=true
Created default portal listening on all IPs (0.0.0.0), port 3260.
/iscsi> ls
o- iscsi ....................................................... [Targets: 1]
  o- iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660 . [TPGs: 1]
    o- tpg1 ..................................... [no-gen-acls, no-auth]
      o- acls ................................................... [ACLs: 0]
      o- luns ................................................... [LUNs: 0]
      o- portals ............................................. [Portals: 1]
        o- 0.0.0.0:3260 ..............................................  [OK]
```

2. Association entre unité logique et portail iSCSI.
   Les numéros d'unités logiques SCSI ou LUNs sont affectés automatiquement. Ici, l'unité `lun0` correspond à la première association faite depuis le dépôt des volumes de stockage.

```
/iscsi> cd iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660/tpg1/luns
/iscsi/iqn.20...660/tpg1/luns> create /backstores/block/blockvol0
Created LUN 0.
/iscsi/iqn.20...660/tpg1/luns> ls
o- luns ..................................................... [LUNs: 1]
  o- lun0 ................. [block/blockvol0 (/dev/vdb) (default_tg_pt_gp)]
```

3. Configuration réseau du portail iSCSI.
   Un même portail peut être en écoute sur IPv4 et IPv6. Dans l'exemple ci-dessous on ouvre une configuration double pile en désignant la totalité des réseaux IPv6 après voir effacé l'entrée créée automatiquement lors de la création du portail.

```
/iscsi/iqn.20...660/tpg1/luns> cd ../portals/
/iscsi/iqn.20.../tpg1/portals> delete 0.0.0.0 3260
Deleted network portal 0.0.0.0:3260
/iscsi/iqn.20.../tpg1/portals> create ::0
Using default IP port 3260
Created network portal ::0:3260.
/iscsi/iqn.20.../tpg1/portals> ls
o- portals ................................................... [Portals: 1]
  o- [::0]:3260 ..................................................... [OK]
```

   On peut sortir de l'outil targetcli pour vérifier que le service réseau est bien accessible. La configuration est sauvegardée automatiquement.

```
/iscsi/iqn.20.../tpg1/portals> exit
Global pref auto_save_on_exit=true
Configuration saved to /etc/rtslib-fb-target/saveconfig.json
$
```

Q35. Comment vérifier la disponibilité du portail réseau iSCSI ?

À l'aide des commandes ss ou lsof, relever le numéro de port de la couche transport relatif au protocole iSCSI.

Sur le système ***initiator***, lancer l'opération de découverte des volumes du portail iSCSI.

Voici un exemple d'exécution de la commande ss depuis le système ***target***.

```
$ ss -tan '( sport = :3260 )'
State       Recv-Q  Send-Q            Local Address:Port        Peer Address:Port         Process
LISTEN      0       256                           *:3260                   *:*
```

Sachant que le service est disponible, on peut utiliser la fonction de découverte sur le système ***initiator***.

```
$ sudo iscsiadm -m discovery --type sendtargets --portal=[2001:678:3fc:171:baad:caff:fefe:5]
[2001:678:3fc:171:baad:caff:fefe:5]:3260,1 iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660
```

Q36. Est-il possible d'ouvrir une session iSCSI à ce stade de la configuration ?

Sur le système ***initiator***, lancer l'opération d'ouverture de session.

Même si le service réseau et la fonction découverte sont ouverts, le volume de stockage réseau n'est pas encore accessible. L'ouverture de session depuis l'hôte ***initiator*** échoue et on obtient le message suivant.

La réponse à la question est donc ***non***.

```
$ sudo iscsiadm -m node \
 -T iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660 \
 -p [2001:678:3fc:171:baad:caff:fefe:5] \
 -l
Logging in to [iface: default,
   target: iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660,
   portal: 2001:678:3fc:171:baad:caff:fefe:5,3260]
iscsiadm: Could not login to [iface: default,
   target: iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660,
   portal: 2001:678:3fc:171:baad:caff:fefe:5,3260].
iscsiadm: initiator reported error (24 - iSCSI login failed due to authorization failure)
iscsiadm: Could not log into all portals
```

Côté hôte ***target***, les journaux système font apparaître un message du type suivant.

```
$ journalctl -n 20 -f --grep scsi
iSCSI Initiator Node: iqn.1993-08.org.debian:01:2cc8dac75cec is not authorized to access iSCSI target portal group: 1.
iSCSI Login negotiation failed.
```

Q37. Comment autoriser l'accès au volume de stockage depuis l'hôte ***initiator*** sans authentication ?

Rechercher les paramètres relatifs à la rubrique `acls` de l'outil targetcli.

Pour que le portail iSCSI accepte l'ouverture d'un session, il est nécessaire de créer une liset de contrôle d'accès avec l'identité du système *initiator*.

Côté *initiator*, on affiche l'identité iSCSI définie lors de l'installation du paquet open-iscsi.

```
$ sudo grep -v ^# /etc/iscsi/initiatorname.iscsi
InitiatorName=iqn.1993-08.org.debian:01:2cc8dac75cec
```

Côte *target*, on créé une nouvelle entrée dans la rubrique `acls` du portail iSCSI via l'outil targetcli.

```
$ sudo targetcli
targetcli shell version 2.1.53
Copyright 2011-2013 by Datera, Inc and others.
For help on commands, type 'help'.

/> cd iscsi/iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660/tpg1/acls
/iscsi/iqn.20...660/tpg1/acls> create iqn.1993-08.org.debian:01:2cc8dac75cec
Created Node ACL for iqn.1993-08.org.debian:01:2cc8dac75cec
Created mapped LUN 0.
/iscsi/iqn.20...660/tpg1/acls>
```

Enfin, en reprenant la commande d'ouverture de session sur le système *initiator*, l'opération est un succès.

```
$ sudo iscsiadm -m node \
 -T iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660 \
 -p [2001:678:3fc:171:baad:caff:fefe:5] \
 -l
Logging in to [iface: default,
  target: iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660,
  portal: 2001:678:3fc:171:baad:caff:fefe:5,3260]
Login to [iface: default,
  target: iqn.2003-01.org.linux-iscsi.target-vm.x8664:sn.bc4899490660,
  portal: 2001:678:3fc:171:baad:caff:fefe:5,3260] successful.
```

À partir de cette étape, le système *initiator* dispose d'une nouvelle unité de stockage en mode bloc.

7. Configuration de l'authentification CHAP

Dans cette partie, on suppose que tous les tests précédents ont été effectués avec succès et que les échanges entre les systèmes *target* et *initiator* sont validés.

On s'intéresse maintenant à l'authentification entre ces mêmes systèmes. Pour traiter les questions suivantes, une nouvelle entrée a été utilisée pour le rôle *target*.

Le mécanisme d'authentification le plus communément utilisé dans le déploiement des connexions iSCSI s'appuie sur CHAP (*Challenge-Handshake Authentication Protocol*). Il s'agit d'une méthode d'authentification entre deux hôtes pairs sans échange de mot de passe en clair sur le réseau. Cette méthode suppose que les deux hôtes utilisent le même mot de passe.

Q38. Comment régler les paramètres d'authentification CHAP sur le système *target* ?

Comme pour les étapes précédentes, toutes les manipulations se font à partir de l'outil targetcli.

Partant d'une nouvelle configuration, on obtient la liste de paramètres suivante dans laquelle aucun contrôle d'accès n'a été défini.

```
/iscsi/iqn.20....5208bcd92edd> ls
o- iqn.2003-01.org.linux-iscsi.target.x8664:sn.5208bcd92edd ..[TPGs: 1]
  o- tpg1 ..........................................[no-gen-acls, no-auth]
    o- acls ................................................ [ACLs: 0]
    o- luns ................................................ [LUNs: 1]
    | o- lun0 ......... [block/blockvol0 (/dev/vdb) (default_tg_pt_gp)]
    o- portals .............................................. [Portals: 1]
      o- [::0]:3260 ............................................. [OK]
```

On passe à la création d'une entrée de contrôle d'accès basée sur l'identifiant `iqn` unique du système *initiator*.

```
/iscsi/iqn.20....5208bcd92edd> create iqn.1993-08.org.debian:01:8c6ecf84c11e
```

```
/iscsi/iqn.20....5208bcd92edd> ls
o- iqn.2003-01.org.linux-iscsi.target.x8664:sn.5208bcd92edd ..[TPGs: 1]
  o- tpg1 ..........................................[no-gen-acls, no-auth]
    o- acls ..................................................[ACLs: 1]
    | o- iqn.1993-08.org.debian:01:8c6ecf84c11e .......[Mapped LUNs: 1]
    |   o- mapped_lun0 ....................[lun0 block/blockvol0 (rw)]
    o- luns ..................................................[LUNs: 1]
    | o- lun0 ..........[block/blockvol0 (/dev/vdb) (default_tg_pt_gp)]
    o- portals ...............................................[Portals: 1]
      o- [::0]:3260 ..............................................[OK]
```

On définit ensuite les paramètres d'authentification pour cette entrée. Comme la méthode CHAP est symétrique, on doit déposer de part et d'autre le secret. On fixe ici les paramètres `userid` et `password`.

```
/iscsi/iqn.20...57c35b07/tpg1> acls/iqn.2015-09.org.debian:01:9d11913c78ac/ set auth userid=SAN-lab-initiator
Parameter userid is now 'SAN-lab-initiator'.
/iscsi/iqn.20...57c35b07/tpg1> acls/iqn.2015-09.org.debian:01:9d11913c78ac/ set auth password=SAN-lab-initiator-53cr3t
Parameter password is now 'SAN-lab-initiator-53cr3t'.
```

Q39. Comment régler les paramètres d'authentification CHAP sur le système *initiator* ?

Rechercher dans le fichier de configuration principal du rôle *initiator* les paramètres relatifs à l'authentification.

Le nom d'utilisateur et le mot de passe sont définis dans le fichier `/etc/iscsi/iscsid.conf` du système *initiator*.

```
# *************
# CHAP Settings
# *************

# To enable CHAP authentication set node.session.auth.authmethod
# to CHAP. The default is None.
node.session.auth.authmethod = CHAP

# To set a CHAP username and password for initiator
# authentication by the target(s), uncomment the following lines:
node.session.auth.username = SAN-lab-initiator
node.session.auth.password = SAN-lab-initiator-53cr3t
```

Le même principe peut être appliqué au mécanisme de découverte en appliquant un couple ***login/password*** identique ou non à la suite de ce fichier de configuration.

Une fois la configuration en place, on obtient les résultats suivants lors de la validation.

- Découverte du nouveau volume réseau :

```
$ sudo iscsiadm -m discovery --type sendtargets --portal=[2001:db8:feb2:2:b8ad:ff:feca:fe00]:3260
[2001:db8:feb2:2:b8ad:ff:feca:fe00]:3260,1 iqn.2003-01.org.linux-iscsi.target.i686:sn.f58f71d5ba26
192.0.2.12:3260,1 iqn.2003-01.org.linux-iscsi.target.i686:sn.f58f71d5ba26
[2001:db8:feb2:2:b8ad:ff:feca:fe00]:3260,1 iqn.2003-01.org.linux-iscsi.target.i686:sn.8b7457c35b07
```

- Connexion avec authentification CHAP :

```
# iscsiadm -m node -T iqn.2003-01.org.linux-iscsi.target.i686:sn.8b7457c35b07 -p 2001:db8:feb2:2:b8ad:ff:feca:fe00 --login
Logging in to [iface: default, target: iqn.2003-01.org.linux-iscsi.target.i686:sn.8b7457c35b07, portal: 2001:db8:feb2:2:b8ad:ff
Login to [iface: default, target: iqn.2003-01.org.linux-iscsi.target.i686:sn.8b7457c35b07, portal: 2001:db8:feb2:2:b8ad:ff:feca
```

- Affichage de la session active :

```
# iscsiadm -m session
tcp: [4] [2001:db8:feb2:2:b8ad:ff:feca:fe00]:3260,1 iqn.2003-01.org.linux-iscsi.target.i686:sn.8b7457c35b07 (non-flash)
```

8. Configuration d'une unité logique RAID1

Dans cette partie, on crée une unité logique RAID1 composée d'une unité de disque locale et d'une unité de disque iSCSI dans le but d'illustrer une solution de réplication synchrone. En effet, dans un volume RAID1 chaque disque contient à tout moment exactement les mêmes données. Ici, le contenu de l'unité de disque locale est identique à celui de l'unité de disque réseau. La réplication ainsi réalisée est dite synchrone puisque toute écriture locale est dupliquée sur le réseau de stockage iSCSI.

8.1. Sélection du paquet et création de l'unité de stockage

Q40. Quel est le paquet qui contient les outils de configuration et de gestion des différents types d'unités RAID logicielles ? Installer ce paquet et identifier l'outil d'administration de tableau RAID logiciel.

Effectuer une recherche dans les descriptions de paquets avec l'acronyme clé RAID.

```
$ aptitude search ~draid | grep administration
p   mdadm   - outil d'administration d'ensembles RAID

$ sudo apt install mdadm
```

Une fois le paquet identifié et installé, on peut lister son contenu et isoler les commandes utilisateur.

```
$ dpkg -L mdadm | grep bin
/sbin
/sbin/mdmon
/sbin/mdadm-startall
/sbin/mdadm
```

Q41. Rechercher la syntaxe d'appel à l'outil identifié dans la question précédente pour créer l'unité logique RAID1 ? Exécuter cette commande.

Après s'être assuré qu'aucune table de partition n'existe sur les deux unités constituant le tableau, on obtient le résultat suivant.

```
$ sudo mdadm --create /dev/md0 --level=raid1 --raid-devices=2 /dev/sda /dev/vdb
mdadm: Note: this array has metadata at the start and
    may not be suitable as a boot device.  If you plan to
    store '/boot' on this device please ensure that
    your boot-loader understands md/v1.x metadata, or use
    --metadata=0.90
Continue creating array? y
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
```

8.2. Manipulations sur l'unité de stockage RAID1

Q42. Comment connaître l'état de l'unité logique RAID1 ?

Effectuer une recherche dans le système de fichiers virtuel `/proc/`.

Exemple du tableau créé lors l'exécution de la commande de la question précédente.

```
$ cat /proc/mdstat
Personalities : [raid1]
md0 : active raid1 vdb[1] sda[0]
      33537920 blocks super 1.2 [2/2] [UU]

unused devices: <none>
```

Q43. Comment afficher la liste des propriétés de l'unité logique RAID1 ?

Effectuer une recherche dans les options de la commande d'administration.

```
$ sudo mdadm --detail /dev/md0
/dev/md0:
           Version : 1.2
     Creation Time : Sat Sep  3 18:07:32 2022
        Raid Level : raid1
        Array Size : 33520640 (31.97 GiB 34.33 GB)
     Used Dev Size : 33520640 (31.97 GiB 34.33 GB)
      Raid Devices : 2
     Total Devices : 2
       Persistence : Superblock is persistent

       Update Time : Sat Sep  3 18:09:18 2022
             State : clean, resyncing
    Active Devices : 2
   Working Devices : 2
    Failed Devices : 0
     Spare Devices : 0

Consistency Policy : resync

     Resync Status : 65% complete

              Name : initiator:0  (local to host initiator)
              UUID : e3da1d56:9df89f79:866d5607:eeb2beff
            Events : 11

    Number   Major   Minor   RaidDevice State
       0       8        0        0      active sync   /dev/sda
       1     254       16        1      active sync   /dev/vdb/
```

Q44. Comment rendre la configuration du tableau RAID1 permanente au niveau système ?

Effectuer une recherche dans les options de la commande d'administration.

C'est le fichier `/etc/mdadm/mdadm.conf` qui contient les directives de configuration. On ajoute en fin de ce fichier la définition du tableau créé plus haut.

```
$ sudo mdadm --detail --scan | sudo tee -a /etc/mdadm/mdadm.conf
```

9. Configuration d'un volume logique et de sa sauvegarde

L'objectif de cette partie est de créer un mécanisme de sauvegarde réseau automatisé en s'appuyant sur la notion de «prise de vue» ou ***snapshot*** proposée par le gestionnaire de volume logique LVM. Dans une prise de vue instantanée, on ne stocke que les différences relativement au volume logique original.

Q45. Quel est le paquet associé à la gestion de volume logique LVM ?

Rechercher et installer le paquet qui permet de créer et gérer des volumes physiques, logiques ainsi que des groupes.

En anglais, on parle de ***Logical Volume Manager*** ou LVM. On cherche donc un paquet avec la chaîne 'lvm'.

```
$ aptitude search ^lvm
p   lvm2           - gestionnaire de volumes logiques de Linux
p   lvm2-dbusd     - démon D-Bus pour LVM2
p   lvm2-lockd     - démon de verrouillage pour LVM
```

```
$ sudo apt install lvm2
```

Q46. Comment créer un volume physique associé au tableau RAID1 précédemment créé ?

Rechercher dans la liste des outils ceux correspondant à la gestion de volume physique.

L'instruction de recherche habituelle est de la forme :

```
$ dpkg -L lvm2 | grep bin
```

Ce sont les outils dont le nom commence par 'pv' qui servent à manipuler les volumes physiques.

```
$ sudo pvcreate --help
```

Création du volume physique.

```
$ sudo pvcreate /dev/md0
  Physical volume "/dev/md0" successfully created.
```

Affichage résumé de l'état du volume physique.

```
$ sudo pvs
  PV         VG Fmt  Attr PSize   PFree
  /dev/md0      lvm2 ---  <31,97g <31,97g
```

Affichage détaillé de l'état du volume physique.

```
$ sudo pvdisplay
  "/dev/md0" is a new physical volume of "<31,97 GiB"
  --- NEW Physical volume ---
  PV Name               /dev/md0
  VG Name
  PV Size               <31,97 GiB
  Allocatable           NO
  PE Size               0
  Total PE              0
  Free PE               0
  Allocated PE          0
  PV UUID               vUlk3p-dzZJ-MyLZ-hMcU-P9dH-oQuB-2lptum
```

Q47. Comment créer un groupe de volume contenant le tableau RAID1 ?

Rechercher dans la liste des outils ceux correspondant à la gestion de groupes de volumes.

À partir du résultat de la commande de recherche de la question précédente, on relève que ce sont les outils dont le nom commence par 'vg' qui servent à manipuler les groupes de volumes.

```
$ sudo vgcreate --help
```

Création du groupe de volume avec un unique volume physique.

```
$ sudo vgcreate lab-vg /dev/md0
  Volume group "lab-vg" successfully created
```

Affichage résumé de l'état du volume physique.

```
$ sudo vgs
  VG         #PV #LV #SN Attr   VSize  VFree
  lab-vg   1   0   0 wz--n- 31,96g 31,96g
```

Affichage détaillé de l'état du volume physique.

```
$ sudo vgdisplay
  --- Volume group ---
  VG Name               lab-vg
  System ID
  Format                lvm2
  Metadata Areas        1
  Metadata Sequence No  1
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                0
  Open LV               0
  Max PV                0
  Cur PV                1
  Act PV                1
  VG Size               31,96 GiB
  PE Size               4,00 MiB
  Total PE              8183
  Alloc PE / Size       0 / 0
  Free  PE / Size       8183 / 31,96 GiB
  VG UUID               KIq2zb-emxQ-JiT0-6wAk-tPlO-MmrN-wxzkLl
```

Q48. Comment créer un volume logique à l'intérieur du groupe contenant le tableau RAID1 ?

Rechercher dans la liste des outils ceux correspondant à la gestion des volumes logiques.

Dans cet exemple, nous allons créer un volume logique de 16Go pour une capacité de 32Go. En situation réelle, il faudrait remplacer les gigaoctets par des téraoctets.

Toujours à partir du résultat de la commande de recherche des deux questions précédentes, on relève que ce sont les outils dont le nom commence par 'lv' qui servent à manipuler les volumes logiques.

```
$ sudo lvcreate --help
```

Création du volume logique de 16Go.

```
$ sudo lvcreate --size 16Go lab-vg
  Logical volume "lvol0" created.
```

Affichage résumé de l'état du volume logique.

```
$ sudo lvs
  LV    VG         Attr       LSize  Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  lvol0 lab-vg -wi-a----- 16,00g
```

Affichage détaillé de l'état du volume logique.

```
$ sudo lvdisplay
  --- Logical volume ---
  LV Path                /dev/lab-vg/lvol0
  LV Name                lvol0
  VG Name                lab-vg
  LV UUID                8UFwye-RMgA-hzY4-8nnv-h7nK-09GA-Ff2AT2
  LV Write Access        read/write
  LV Creation host, time initiator, 2022-09-05 14:27:28 +0200
  LV Status              available
  # open                 0
  LV Size                16,00 GiB
  Current LE             4096
  Segments               1
  Allocation             inherit
  Read ahead sectors     auto
  - currently set to     256
  Block device           252:0
```

Q49. Comment créer un système de fichiers sur le nouveau volume logique ?

Reprendre les traitements de la Section 4, « Préparer une unité de stockage » avec le nom du volume logique obtenu à la question précédente.

Formatage du système de fichiers.

```
$ sudo mkfs.ext4 /dev/lab-vg/lvol0
mke2fs 1.46.5 (30-Dec-2021)
Discarding device blocks: done
Creating filesystem with 4194304 4k blocks and 1048576 inodes
Filesystem UUID: d94a59f2-7c67-4be7-9643-d646d1dbe2a4
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

Q50. Comment monter et accéder au nouveau système de fichiers ?

Créer un sous dossier au niveau `/mnt` et monter le nouveau système de fichiers manuellement.

Exemple de résultats attendus.

```
$ sudo mkdir /mnt/lvol0

$ sudo mount /dev/lab-vg/lvol0 /mnt/lvol0/

$ mount | grep lvol0
/dev/mapper/lab--vg-lvol0 on /mnt/lvol0 type ext4 (rw,relatime)
```

Une fois le système de fichiers monté, il est possible de créer des dossiers et des fichiers avec les permissions adaptées. Voici un exemple avec une attribution de dossier à l'utilisateur normal `etu`.

```
$ sudo mkdir /mnt/lvol0/etu-files
$ sudo chown etu.etu /mnt/lvol0/etu-files
$ touch /mnt/lvol0/etu-files/my-first-file
```

Q51. Comment visualiser l'état global des systèmes de fichiers et des montages en cours ?

Utiliser les commandes usuelles telles que df et lsblk.

Exemple de résultat attendu.

```
$ df -hT
Sys. de fichiers          Type     Taille Utilisé Dispo Uti% Monté sur
udev                      devtmpfs   463M       0  463M   0% /dev
tmpfs                     tmpfs       97M    700K   97M   1% /run
/dev/vda2                 ext4       117G    2,0G  109G   2% /
tmpfs                     tmpfs      484M       0  484M   0% /dev/shm
tmpfs                     tmpfs      5,0M       0  5,0M   0% /run/lock
/dev/vda1                 vfat       511M    3,5M  508M   1% /boot/efi
tmpfs                     tmpfs       97M       0   97M   0% /run/user/1000
/dev/mapper/lab--vg-lvol0 ext4        16G     28K   15G   1% /mnt/lvol0
```

```
$ lsblk
NAME              MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINTS
sda                 8:0    0    32G  0 disk
sdb                 8:16   0    32G  0 disk
└─md0               9:0    0    32G  0 raid1
  └─lab--vg-lvol0 252:0    0    16G  0 lvm   /mnt/lvol0
sr0                11:0    1  1024M  0 rom
vda               254:0    0   120G  0 disk
├─vda1            254:1    0   512M  0 part  /boot/efi
├─vda2            254:2    0 118,5G  0 part  /
└─vda3            254:3    0   977M  0 part  [SWAP]
vdb               254:16   0    32G  0 disk
└─md0               9:0    0    32G  0 raid1
  └─lab--vg-lvol0 252:0    0    16G  0 lvm   /mnt/lvol0
```

Cette dernière commande illustre bien l'état de la réplication RAID1 en plus de l'utilisation du volume logique.

Q52. Comment créer deux photos instantanées du volume logique avec des jeux de fichiers différents ?

Après avoir créé une série de fichiers, rechercher les options de la commande lvcreate qui permettent de créer la première prise de vue (***snapshot***).

Création de 10 fichiers vides.

```
$ for i in {1..10}
do
 touch /mnt/lvol0/etu-files/first-$(printf "%02d" $i)-file
done

$ ls -1 /mnt/lvol0/etu-files/
first-01-file
first-02-file
first-03-file
first-04-file
first-05-file
first-06-file
first-07-file
first-08-file
first-09-file
first-10-file
my-first-file
```

Première capture instantanée du système de fichiers.

```
$ sudo lvcreate --snapshot --name fisrt-snap -L 500M /dev/lab-vg/lvol0
  Logical volume "fisrt-snap" created.
```

Création de 10 nouveaux fichiers vides.

```
$ for i in {1..10}
do
    touch /mnt/lvol0/etu-files/second-$(printf "%02d" $i)-file
done

$ ls -1 /mnt/lvol0/etu-files/
first-01-file
first-02-file
first-03-file
first-04-file
first-05-file
first-06-file
first-07-file
first-08-file
first-09-file
first-10-file
my-first-file
second-01-file
second-02-file
second-03-file
second-04-file
second-05-file
second-06-file
second-07-file
second-08-file
second-09-file
second-10-file
```

Seconde capture instantanée du système de fichiers.

```
$ sudo lvcreate --snapshot --name second-snap -L 500M /dev/lab-vg/lvol0
  Logical volume "second-snap" created.
```

État du volume logique.

```
$ sudo lvs
LV          VG     Attr       LSize   Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  fisrt-snap  lab-vg swi-a-s--- 500,00m      lvol0  0,01
  lvol0       lab-vg owi-aos---  16,00g
  second-snap lab-vg swi-a-s--- 500,00m      lvol0  0,01
```

Q53. Comment tester la restauration du système de fichiers à partir des instantanés ?

Après avoir supprimé tous les fichiers du dossier `/mnt/lvol0/etu-files/`, on restaure le contenu des deux prises de vues dans l'ordre.

Suppression des fichiers du répertoire de travail.

```
$ rm /mnt/lvol0/etu-files/*
```

Restauration à partir du premier instantané.

```
$ sudo lvconvert --merge /dev/lab-vg/fisrt-snap
  Delaying merge since origin is open.
  Merging of snapshot lab-vg/fisrt-snap will occur on next activation of lab-vg/lvol0.
```

Pour que la restauration soit effective, il est nécessaire de désactiver/réactiver le volume logique à l'aide de la commande lvchange.

```
$ sudo lvchange --activate n lab-vg/lvol0
  Logical volume lab-vg/lvol0 is used by another device.
```

Aïe ! Le volume logique est en cours d'utilisation. On doit donc démonter le système de fichiers et tester à nouveau.

```
$ sudo umount /mnt/lvol0
$ sudo lvchange --activate n lab-vg/lvol0
```

Cette fois ci, le volume est enfin désactivé. On peut le réactiver.

```
$ sudo lvchange --activate y lab-vg/lvol0
```

```
$ sudo lvscan
  ACTIVE   Original '/dev/lab-vg/lvol0' [16,00 GiB] inherit
  ACTIVE   Snapshot '/dev/lab-vg/second-snap' [500,00 MiB] inherit
```

La partition est bien disponible et on a retrouvé la liste des fichiers du premier instantané.

```
$ sudo mount /dev/lab-vg/lvol0 /mnt/lvol0/

$ ls -1 /mnt/lvol0/etu-files/
first-01-file
first-02-file
first-03-file
first-04-file
first-05-file
first-06-file
first-07-file
first-08-file
first-09-file
first-10-file
my-first-file
```

Pour restaurer le contenu du second instantané, il faut reprendre les mêmes opérations à partir de la commande lvconvert.

10. Perte d'une unité de disque du tableau RAID1

L'objectif de cette partie est de simuler la perte d'une unité de disque du tableau RAID1 et de provoquer la reconstruction de ce tableau depuis l'unité de disque réseau iSCSI. On illustre ainsi le mécanisme de tolérance aux pannes en plus de l'utilisation des ***snapshots*** du gestionnaire de volumes logiques LVM.

Q54. Comment provoquer une panne de disque côté *initiator* ?

1. Extinction de la machine virtuelle avec le rôle *initiator*.
2. Suppression du fichier image du disque supplémentaire de la machine virtuelle.
3. Redémarrage de la même machine virtuelle.

11. Évaluation des performances

La pertinence ou la validité des résultats obtenus avec la commande sysbench dépendent énormément du facteur temps. Une mesure valide suppose un temps d'exécution de quelques heures au moins. Les résultats donnés ici ne sont que des échantillons.

```
$ sudo apt install sysbench
```

Unité de disque locale

Système de fichiers `ext4`.

```
$ mkdir /var/tmp/benchmark

$ cd /var/tmp/benchmark/
```

```
$ sysbench fileio prepare
```

```
$ sysbench fileio --file-test-mode=rndrw run
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
Read/Write ratio for combined random IO test: 1.50
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing random r/w test
Initializing worker threads...

Threads started!


File operations:
    reads/s:                      6062.94
    writes/s:                     4041.90
    fsyncs/s:                     12939.18

Throughput:
    read, MiB/s:                  94.73
    written, MiB/s:               63.15

General statistics:
    total time:                          10.0062s
    total number of events:              230569

Latency (ms):
         min:                                    0.00
         avg:                                    0.04
         max:                                  133.67
         95th percentile:                        0.15
         sum:                                 9943.24

Threads fairness:
    events (avg/stddev):           230569.0000/0.00
    execution time (avg/stddev):   9.9432/0.00
```

Volume logique LVM sur une unité de disque RAID1 avec un membre iSCSI

Système de fichiers `ext4`.

```
$ mkdir /mnt/lvol0/etu-files/benchmark

$ cd /mnt/lvol0/etu-files/benchmark
```

```
$ sysbench fileio prepare
```

```
$ sysbench fileio --file-test-mode=rndrw run
sysbench 1.0.20 (using system LuaJIT 2.1.0-beta3)

Running the test with following options:
Number of threads: 1
Initializing random number generator from current time


Extra file open flags: (none)
128 files, 16MiB each
2GiB total file size
Block size 16KiB
Number of IO requests: 0
Read/Write ratio for combined random IO test: 1.50
Periodic FSYNC enabled, calling fsync() each 100 requests.
Calling fsync() at the end of test, Enabled.
Using synchronous I/O mode
Doing random r/w test
Initializing worker threads...

Threads started!

File operations:
    reads/s:                      1309.93
    writes/s:                     873.29
    fsyncs/s:                     2799.10

Throughput:
    read, MiB/s:                  20.47
    written, MiB/s:               13.65

General statistics:
    total time:                          10.0295s
    total number of events:              49850

Latency (ms):
         min:                                    0.00
         avg:                                    0.20
         max:                                   31.26
         95th percentile:                        0.59
         sum:                                 9978.89

Threads fairness:
    events (avg/stddev):           49850.0000/0.00
    execution time (avg/stddev):   9.9789/0.00
```

12. Documents de référence

Architecture réseau des travaux pratiques

Infrastructure : présentation de l'implantation des équipements d'interconnexion réseau dans l'armoire de brassage et du plan d'adressage IP prédéfini pour l'ensemble des séances de travaux pratiques.

Configuration d'une interface réseau

Configuration d'une interface de réseau local : tout sur la configuration des interfaces réseau de réseau local.

iSCSI - Debian Wiki

La page *iSCSI and Debian* contient deux sous-rubriques sur les rôles ***initiator*** et ***target***.

Introduction au système de fichiers réseau NFSv4

https://www.inetdoc.net

Résumé

L'objectif de ce support de travaux pratiques est l'étude du système de fichiers réseau NFS. Il illustre les accès en «mode fichier» à une unité de stockage réseau. Ce mode d'accès correspond à un stockage de type NAS ou *Network Attached Storage*. Le document débute avec l'étude du principe de fonctionnement des appels de fonctions RPC (*Remotre Procedure Call*) puis il poursuit avec la configuration d'un serveur NFS qui exporte une arborescence de comptes utilisateurs. Côté client, on étudie les accès au système de fichiers réseau NFS suivant deux modes distincts : le montage manuel puis l'automontage.

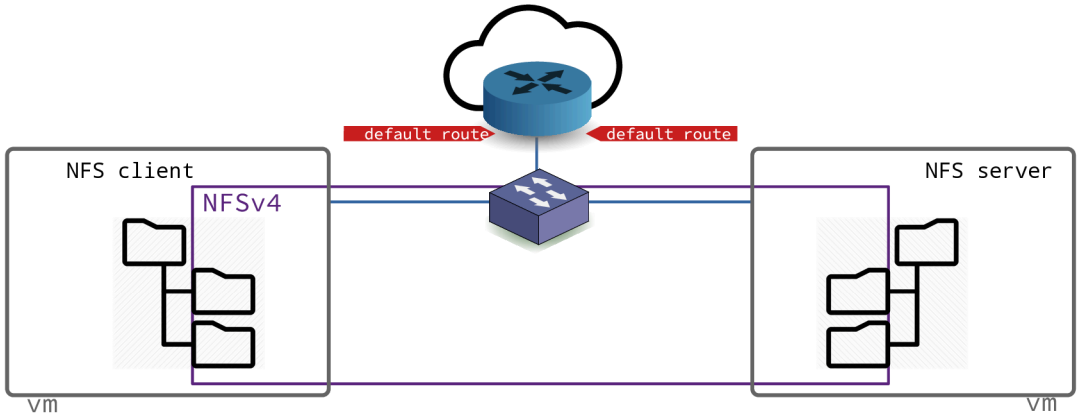
Table des matières

1. Topologie, scénario et plan d'adressage

Topologie logique

Les manipulations présentées dans ce support utilisent un domaine de diffusion unique (VLAN) dans lequel on trouve au moins deux systèmes virtuels ou physiques avec deux rôles distincts.

- Le système *serveur exporte* une arborescence de son système de fichiers local à destination des clients.
- Le(s) système(s) *client(s) montent* le système de fichiers réseau sur une arborescence locale.



Topologie logique - vue complète

Scénario

L'objectif des manipulations demandées dans ce document est d'illustrer les fonctionnalités apportées par le protocole NFS. Le séquencement des opérations à réaliser lors de la séance de travaux pratiques est décrit dans le tableau ci-dessous. Après le traitement de la première partie commune, les deux postes occupent chacun un rôle distinct.

Tableau 1. Attribution des rôles NFS

Client	Serveur
Identification du mécanisme des appels RPC. Installation et configuration des paquets communs.	
Identification des services disponibles sur le serveur. Création d'un compte local sans répertoire utilisateur.	Installation du paquet spécifique au serveur et configuration du service en fonction de l'arborescence à exporter.
validation de l'accès au système de fichiers réseau avec capture de trafic.	
Installation du paquet spécifique et configuration du service d'automontage des répertoires utilisateurs.	

Pour ces travaux pratiques, de nombreuses questions peuvent être traitées à l'aide du document de référence : *Nfsv4 configuration*. Il faut cependant faire correspondre les configurations décrites dans ce document avec les configurations proposées avec les paquets de la distribution ***Debian GNU/Linux***.

Plan d'adressage

Partant de la topologie présentée ci-dessus, on utilise un plan d'adressage pour chacun des rôles iSCSI.

Le tableau ci-dessous correspond au plan d'adressage de la maquette qui a servi à traiter les questions des sections suivantes. Lors des séances de travaux pratiques, un plan d'adressage spécifique est fourni à chaque binôme d'étudiants. Il faut se référer au document *Infrastructure*.

Tableau 2. Plan d'adressage de la maquette « Introduction au système de fichiers réseau NFSv4 »

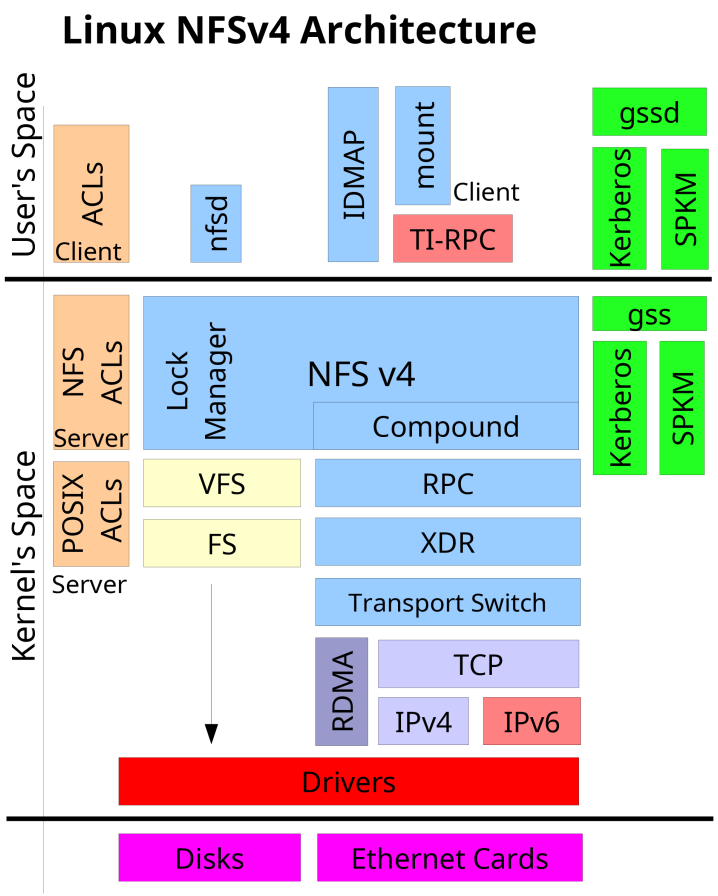
Rôle	VLAN	Adresses IP	Interface tap
Client NFS	501	`192.168.51.194/27` `2001:678:3fc:1f5::195/64`	2
Serveur NFS	501	`192.168.51.195/27` `2001:678:3fc:1f5::195/64`	3

Avant de traiter les questions des sections suivantes, il faut rechercher dans le document *Infrastructure* les éléments nécessaires au raccordement des machines virtuelles ou physiques.

2. Protocole NFS

Cette section reprend les éléments spécifiques au protocole NFS introduits lors de la présentation *Systèmes de fichiers réseau*.

Plusieurs versions du protocole de système de fichiers réseau NFS sont disponibles. Chacune correspond à une «époque» ou à un mode d'exploitation. La vue ci-dessous illustre la distribution des fonctionnalités de la version 4 entre les espaces noyau et utilisateur.



La version 2 du protocole NFS a été la première à être largement adoptée à la fin des années 80. Elle a été conçue pour fournir un service de partage de fichiers entre les hôtes d'un même réseau local. Elle s'appuie sur le protocole UDP au niveau transport et sur le mécanisme d'appel de procédure distant (RPC) aux niveaux supérieurs.

La version 3 du protocole, introduite au milieu des années 90, a apporté de nombreuses améliorations en termes de fiabilité et de performances relativement à la précédente. Avec la version 3 du protocole :

- La taille maximum de fichier n'est plus limitée à 2Go.
- Les écritures asynchrones sur le serveur sont possibles ; ce qui améliore beaucoup les performances. Les requêtes en écriture des clients sont gérées en mémoire cache. Le client n'a plus à attendre que les demandes d'écritures soient effectivement appliquées sur les disques ce qui améliore les temps de réponse.
- Les contrôles d'accès sont effectués avant les manipulations sur les fichiers.
- La taille des données transférées n'est plus limitée à 8Ko.
- Il est possible d'utiliser le protocole TCP au niveau transport.

La version 4 du protocole apporte de nouvelles fonctionnalités relativement aux précédentes.

Les identifiants d'utilisateur et de groupe (`uid`/`gid`) sont représentés par des chaînes de caractères. Un service, baptisé `idmapd`, est utilisé sur le serveur pour faire les correspondances entre les valeurs numériques locales et les chaînes de caractères. Ces correspondances permettent d'utiliser de nouveaux contrôles d'accès indépendants entre clients et serveurs.

Les serveurs maintiennent un pseudo système de fichiers qui assure la cohérence du système de nommage avec les clients. Ainsi, un objet est nommé de façon identique entre le serveur et ses clients. Pour respecter les spécifications POSIX, un client qui a accès à un niveau d'arborescence peut parcourir tous les niveaux inférieurs. Il n'est pas nécessaire d'exporter les sous arborescences.

Les appels de procédures distants n'utilisent plus le multiplexage de ports. Un numéro de port unique a été attribué à la version 4 du protocole NFS : tcp/2049. La version 3 doit utiliser plusieurs ports pour les traitements de ses protocoles complémentaires ; ce qui donne un assemblage plutôt complexe de ports et de couches avec des problèmes de sécurité propres. Aujourd'hui, ce mode de fonctionnement est abandonné et toutes les opérations de mise en œuvre de protocole complémentaire précédemment exécutées via des ports individuels sont maintenant traitées directement à partir d'un port unique connu.

Désormais, le mécanisme d'appel RPC n'est plus aussi important et sert essentiellement d'enveloppe pour les opérations encapsulées dans la pile NFSv4. Ce changement rend le protocole beaucoup moins dépendant de la sémantique du système de fichiers sous-jacent. Pour autant, les opérations de système de fichiers d'autres systèmes d'exploitation n'ont pas été négligées. Par exemple, les systèmes Microsoft™ exigent des appels ***stateful*** ouverts. Le mécanisme de suivi d'état de communication (***statefulness***) facilite l'analyse de trafic et rend les opérations de système de fichiers beaucoup plus simples à interpréter. Ce même mécanisme permet aux clients de gérer les données «en l'état» en mémoire cache.

La version 4 simplifie les requêtes en utilisant des opérations composées ou groupées (***compound***) qui englobent un grand nombre de traitements sur les objets du système de fichiers. L'effet immédiat est, bien sûr, une diminution très importante des appels RPC et des données qui doivent parcourir le réseau. Bien que chaque appel RPC transporte beaucoup plus de données en accomplit beaucoup plus de traitements, on considère qu'une requête composée de la version 4 du protocole exige cinq fois moins d'interactions client serveur qu'avec la version 3.

3. Configuration commune au client et au serveur NFS

Plusieurs services communs doivent être actifs pour que les accès au système de fichiers réseau NFS soient utilisables. Le mécanisme de gestion des appels de procédures distants appelé RPC ou ***Remote Procedure Call*** constitue le point de départ dans la mise œuvre de ces services communs.

Le logiciel de gestion des appels de procédures distants a évolué avec les différentes versions du système de fichiers NFS et l'arrivée du protocole réseau IPv6. La configuration étudiée ici doit permettre de fonctionner de la façon la plus transparente possible avec les versions 3 et 4 du système de fichiers NFS.

> Note
>
> Les manipulations présentées ici ne traitent pas le volet authentification et chiffrement des échanges sur le réseau. On considère que les services ***Kerberos***, SPKM-3 et LIPKEY ne sont pas actifs sur les systèmes étudiés.

3.1. Gestion des appels RPC

Q55. Quels sont les deux logiciels disponibles chargés de la gestion des appels RPC ? Qu'est-ce qui les distinguent ?

La présentation *Systèmes de fichiers réseau* introduit les principes de fonctionnement des appels de procédures distants.

Rechercher dans le support *Linux NFS-HOWTO* le service «historique» utilisé par NFS pour le multiplexage des appels de procédures distants.

Le support *Linux NFS-HOWTO* présente le service «historique» utilisé par NFS pour le multiplexage des appels de procédure distants : `portmap`. Ce service est fourni par le paquet du même nom et est limité au protocole réseau IPv4.

Le démon `rpcbind` actuel est aussi fourni par le paquet du même nom. C'est un logiciel de multiplexage des appels de procédure distants qui se veut plus évolutif que le précédent et qui supporte le protocole réseau IPv6.

Q56. Quel est le paquet qui correspond à la gestion des appels de procédure distants ?.

Utiliser les outils de recherche dans les répertoires de noms de paquets et dans leurs descriptions : apt-cache, dpkg, aptitude.

Comme indiqué dans la documentation, on recherche un paquet portant le nom `rpcbind`.

```
apt search rpcbind
En train de trier... Fait
Recherche en texte intégral... Fait
rpcbind/testing 1.2.6-6+b1 amd64
  conversion de numéros de programmes RPC en adresses universelles
```

```
sudo apt install rpcbind
```

Q57. Quel est le numéro de port utilisé par le service ? Quel est le principe de fonctionnement du service pour le traitement des appels de procédures distants ?

Utiliser les commandes qui permettent d'obtenir les informations sur :

- La liste des processus actifs sur le système,
- Les numéros de ports en écoute sur les interfaces réseau,

- Les pages de manuels des applications utilisées.
- La liste des processus actifs sur le système,

```
ps aux | grep rpc[b]ind
root      2963  0.0  0.0  18956   724 ?        Ss   14:01   0:00 /sbin/rpcbind -w
```

- Les numéros de ports en écoute sur les interfaces réseau,

```
sudo lsof -i | grep rpc[b]ind
rpcbind   2096        _rpc    4u  IPv4  18957      0t0  TCP *:sunrpc (LISTEN)
rpcbind   2096        _rpc    5u  IPv4    713      0t0  UDP *:sunrpc
rpcbind   2096        _rpc    6u  IPv6   1752      0t0  TCP *:sunrpc (LISTEN)
rpcbind   2096        _rpc    7u  IPv6  20601      0t0  UDP *:sunrpc
```

On obtient la correspondance entre numéro de port et nom de service en consultant le fichier `/etc/services`.

```
grep sunrpc /etc/services
sunrpc          111/tcp         portmapper      # RPC 4.0 portmapper
sunrpc          111/udp         portmapper
```

Le principe de fonctionnement des appels de procédure distants veut que tous ces appels soient reçus sur un numéro de port unique : `sunrpc/111`. Ces appels, une fois identifiés, sont transmis aux programmes concernés pour être traités.

- Les pages de manuels des applications utilisées.

```
man rpcbind
```

Q58. Quelle est a commande qui permet de lister les services accessibles via un appel RPC ? À quel paquet appartient cette commande ?

Rechercher dans le support *Linux NFS-HOWTO* et dans la liste des fichiers du paquet sélectionné pour la gestion des appels RPC.

La commande présentée dans le support *Linux NFS-HOWTO* est appelée rpcinfo. On vérifie sa présence sur le système étudié de la façon suivante.

```
dpkg -S $(which rpcinfo)
rpcbind: /usr/sbin/rpcinfo
```

C'est l'option `-s` qui permet d'obtenir la présentation la plus synthétique des services accessibles par appel RPC.

```
rpcinfo -s
   program version(s) netid(s)                         service     owner
    100000  2,3,4     local,udp,tcp,udp6,tcp6          portmapper  superuser
```

La copie d'écran ci-dessus montre que le gestionnaire d'appel `portmapper` est le seul service ouvert. On relève l'ordre de priorité des différentes versions du service supportées par le système ainsi que les versions des protocoles de couche transport.

Q59. Donner deux exemples d'exécution de la commande pour lister le(s) service(s) ouvert sur le système local puis sur le système voisin.

Reprendre la commande utilisée dans la question précédente en indiquant l'adresse IPv4 ou IPv6 du système voisin.

L'exemple d'exécution de la commande en local est donné dans la copie d'écran de la question précédente. Pour connaître les services accessibles sur un autre poste, on utilise la même commande suivie de l'adresse IP de cet hôte.

```
rpcinfo -s 192.168.51.194
   program version(s) netid(s)                         service     owner
    100000  2,3,4     local,udp,tcp,udp6,tcp6          portmapper  superuser
```

```
rpcinfo -s fe80::baad:caff:fefe:2
   program version(s) netid(s)                         service     owner
    100000  2,3,4     local,udp,tcp,udp6,tcp6          portmapper  superuser
```

Ces copies d'écran montrent la même liste de paramètres que lors de l'exécution de la commande en local. Les configurations sur les deux hôtes sont donc identiques à ce stade de la configuration.

Q60. Réaliser une capture à l'aide de l'analyseur réseau lors de l'exécution de la commande et relever : le protocole de transport utilisé, les numéros de ports caractéristiques de cette transaction ainsi que le nom de la procédure RPC utilisée.

```
système 1                                    système 2
-----------------------------------------------------------
 <commande>       --- requête --->           <processus>

                  <-- réponse ----
```

Voici un exemple de capture en mode console qui donne les éléments demandés.

Note

Pour effectuer des captures de trafic réseau en mode console, on dispose de deux applications : tshark et termshark. Pour limiter les dimensions des copies d'écran, on privilégie l'utilisation de tshark.

Pour utiliser l'une ou l'autre des deux applications en tant qu'utilisateur normal, il est nécessaire d'appartenir au groupe `wireshark`. Pour ajouter le compte `etu` au groupe système, on exécute l'instruction `sudo adduser etu wireshark`. Il ne faut pas oublier de se déconnecter puis se reconnecter pour bénéficier de l'attribution au groupe.

Pour une requête IPv4, on obtient :

```
tshark -i enp0s1
Capturing on 'enp0s1'
192.168.51.195 → 192.168.51.194 TCP 74 53284 → 111 [SYN] Seq=0
192.168.51.194 → 192.168.51.195 TCP 74 111 → 53284 [SYN, ACK] Seq=0 Ack=1
192.168.51.195 → 192.168.51.194 TCP 66 53284 → 111 [ACK] Seq=1 Ack=1
192.168.51.195 → 192.168.51.194 Portmap 110 V3 DUMP Call
192.168.51.194 → 192.168.51.195 TCP 66 111 → 53284 [ACK] Seq=1 Ack=45
192.168.51.194 → 192.168.51.195 Portmap 754 V3 DUMP Reply (Call In 4)
192.168.51.195 → 192.168.51.194 TCP 66 53284 → 111 [ACK] Seq=45 Ack=689
192.168.51.195 → 192.168.51.194 TCP 66 53284 → 111 [FIN, ACK] Seq=45 Ack=689
192.168.51.194 → 192.168.51.195 TCP 66 111 → 53284 [FIN, ACK] Seq=689 Ack=46
192.168.51.195 → 192.168.51.194 TCP 66 53284 → 111 [ACK] Seq=46 Ack=690
```

Pour une requête IPv6 avec l'adresse unique, on obtient :

```
tshark -i enp0s1
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 TCP 94 51134 → 111 [SYN] Seq=0
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 94 111 → 51134 [SYN, ACK] Seq=0 Ack=1
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 TCP 86 51134 → 111 [ACK] Seq=1 Ack=1
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 Portmap 130 V3 DUMP Call
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 86 111 → 51134 [ACK] Seq=1 Ack=45
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 Portmap 774 V3 DUMP Reply (Call In 4)
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 TCP 86 51134 → 111 [ACK] Seq=45 Ack=689
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 TCP 86 51134 → 111 [FIN, ACK] Seq=45 Ack=689
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 86 111 → 51134 [FIN, ACK] Seq=689 Ack=46
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 TCP 86 51134 → 111 [ACK] Seq=46 Ack=690
```

- Le protocole de couche transport utilisé est TCP.
- Le numéro de port utilisé correspond bien au service enregistré `sunrpc/111`.
- Le sous-programme distant appelé est : `Portmap V3 DUMP Call`.

Pour une requête IPv6 avec l'adresse de lien local, on obtient :

```
tshark -i enp0s1 -f "! port 22"
Capturing on 'enp0s1'
    1 0.000000000 fe80::baad:caff:fefe:3 → fe80::baad:caff:fefe:2 Portmap 102 V3 DUMP Call
    2 0.000265556 fe80::baad:caff:fefe:2 → fe80::baad:caff:fefe:3 Portmap 746 V3 DUMP Reply (Call In 1)
2 packets captured
```

Ici, le protocole de couche transport utilisé est UDP. Comme UDP est non orienté connexion, on ne relève aucune trace d'ouverture ou de fermeture de connexion.

On remarque que la copie d'écran ci-dessus utilise une syntaxe de capture qui permet de filtrer tous les segments qui font appel au port numéro `22` qui correspond au service SSH.

```
tshark -i enp0s1 -f "! port 22"
```

Pour exploiter toutes les informations du trafic capturé, il est conseillé de stocker les résultats dans un fichier à l'aide de la syntaxe suivante.

```
tshark -i enp0s1 -f "! port 22" -w /var/tmp/rpcbind.pcap
Capturing on 'enp0s1'
3 ^C
```

Dans ce dernier cas, seul le compte des trames capturées apparaît à la console.

On peut alors transférer le fichier de capture via la commande scp pour une exploitation via l'interface graphique de Wireshark ou afficher les détails directement à la console. Dans l'exemple ci-dessous, on affiche toutes les informations relatives à la première trame capturée.

```
tshark -r /var/tmp/rpcbind.pcap -V -Y "frame.number == 1"
```

3.2. Gestion des paquets NFS

Q61. Quel est le paquet commun au client et au serveur ? Identifier le jeu de commandes fournies par ce paquet.

Rechercher dans la liste des paquets disponibles, ceux dont le nom débute par `nfs`.

```
aptitude search ?name"(^nfs)" | grep -v ganesha
v  nfs-client -
p  nfs-common - NFS support files common to client and server
p  nfs-kernel-server - support for NFS kernel server
v  nfs-server -
p  nfs4-acl-tools - Commandline and GUI ACL utilities for the NFSv4 client
p  nfstrace - NFS tracing/monitoring/capturing/analyzing tool
p  nfstrace-doc - NFS tracing/monitoring/capturing/analyzing tool (documentation)
p  nfswatch - Program to monitor NFS traffic for the console
```

Dans la liste ci-dessus, on identifie le paquet `nfs-common` qui correspond bien aux fonctions communes au client et au serveur NFS.

```
sudo apt install nfs-common
```

Une fois le paquet installé, la liste des programmes fournis par ce paquet est extraite de la liste de ses fichiers à l'aide de la commande suivante.

```
dpkg -L nfs-common | grep bin
/sbin
/sbin/mount.nfs
/sbin/osd_login
/sbin/rpc.statd
/sbin/showmount
/sbin/sm-notify
/usr/sbin
/usr/sbin/blkmapd
/usr/sbin/mountstats
/usr/sbin/nfsidmap
/usr/sbin/nfsiostat
/usr/sbin/nfsstat
/usr/sbin/rpc.gssd
/usr/sbin/rpc.idmapd
/usr/sbin/rpc.svcgssd
/usr/sbin/rpcdebug
/usr/sbin/start-statd
/sbin/mount.nfs4
/sbin/umount.nfs
/sbin/umount.nfs4
```

Dans cette liste, on trouve les commandes de montage, de démontage et de suivi d'état du système de fichiers réseau.

4. Configuration du serveur NFS

Le rôle du serveur NFS est de mettre à disposition sur le réseau une partie de son arborescence locale de système de fichiers. On parle d'«exportation».

Note

Il existe plusieurs implémentations libres de serveur NFS. On se limite ici à l'utilisation du logiciel lié au noyau Linux.

Q62. Quel est le paquet qui contient les outils nécessaires au fonctionnement du serveur NFS ? Installez ce paquet.

Interroger les méta données du gestionnaire de paquets pour identifier le nom du paquet à installer.

La recherche des mots clés `nfs` et `server` donne les résultats suivants.

```
aptitude search '?and(nfs, server)'
p   nfs-kernel-server      - support for NFS kernel server
v   nfs-server
```

Les informations données par la commande `apt show nfs-kernel-server` permettent de confirmer qu'il s'agit bien du paquet à installer.

```
sudo apt -y install nfs-kernel-server
```

Q63. Quel est le fichier de configuration principal de gestion des exportations NFS ?

Rechercher dans le support *Linux NFS-HOWTO*.

Quelles que soient les versions du protocole, c'est toujours le fichier `/etc/exports` qui est utilisé. Ce fichier est présenté dans le support *Linux NFS-HOWTO*. Le fichier livré avec le paquet contient, en commentaires, deux exemples complets de configuration NFSv3 et NFSv4. C'est ce dernier exemple que l'on adapte pour traiter les questions suivantes.

Q64. Créer le répertoire `/home/exports/home`. Quelles sont les instructions d'exportation à ajouter au fichier de configuration pour ce répertoire ?

Rechercher dans les supports *Linux NFS-HOWTO* et *Nfsv4 configuration*. On peut aussi utiliser les pages de manuels fournies avec le paquet du serveur NFS.

En exploitant la documentation *Nfsv4 configuration* et l'exemple donné dans le fichier de configuration, on applique les instructions de configuration suivantes dans le fichier `/etc/exports`.

```
sudo mkdir -p /home/exports/home
```

```
cat << EOF | sudo tee -a /etc/exports
/home/exports           192.168.51.192/27(rw,sync,fsid=0,crossmnt,no_subtree_check)
/home/exports/home      192.168.51.192/27(rw,sync,no_subtree_check)
EOF
```

```
cat << EOF | sudo tee -a /etc/exports
/home/exports           2001:678:3fc:1f5::/64(rw,sync,fsid=0,crossmnt,no_subtree_check)
/home/exports/home      2001:678:3fc:1f5::/64(rw,sync,no_subtree_check)
EOF
```

Bien sûr, les adresses des réseaux IPv4 et/ou IPv6 doivent être adaptées au contexte.

Les options entre parenthèses sont documentées dans les pages de manuels `exports` : `man 5 exports`. Les éléments de la liste suivante sont extraits de cette documentation.

- `rw` : autoriser les requêtes en lecture et en écriture sur le volume NFS. Le comportement par défaut est d'interdire toute requête qui modifierait le système de fichiers.
- `sync` : ne répondre aux requêtes qu'après l'exécution de tous les changements sur le support réel.
- `fsid=0` : avec NFSv4, un système de fichiers particulier est la racine de tous les systèmes de fichiers partagés. Il est défini par fsid=root ou fsid=0, qui veulent tous deux dire exactement la même chose.
- `crossmnt` : cette option permet aux clients de se déplacer du système de fichiers marqué `crossmnt` aux systèmes de fichiers partagés montés dessus. Voir l'option nohide.
- `no_subtree_check` : cette option neutralise la vérification de sous-répertoires, ce qui a des subtiles implications au niveau de la sécurité, mais peut améliorer la fiabilité dans certains cas. Si un sous-répertoire dans un système de fichiers est partagé, mais que le système de fichiers ne l'est pas, alors chaque fois qu'une requête NFS arrive, le serveur doit non seulement vérifier que le fichier accédé est dans le système de fichiers approprié (ce qui est facile), mais aussi qu'il est dans l'arborescence partagée (ce qui est plus compliqué). Cette vérification s'appelle subtree_check.

Q65. Comment rendre la configuration d'exportation NFS effective ? Comment vérifier que les paramètres actifs sont corrects ?

Rechercher dans la liste des outils fournis avec le paquet `nfs-kernel-server` la commande qui permet de connaître l'état courant des exportations NFS.

On identifie la commande exportfs dans la liste des binaires fournis avec le paquet serveur NFS.

```
dpkg -L nfs-kernel-server | grep bin
/sbin
/sbin/nfsdcltrack
/usr/sbin
/usr/sbin/exportfs
/usr/sbin/rpc.mountd
/usr/sbin/rpc.nfsd
```

Après chaque modification d'un fichier de configuration, il ne faut surtout pas oublier de relancer le service correspondant.

```
sudo systemctl restart nfs-kernel-server
```

```
systemctl status nfs-kernel-server
# nfs-server.service - NFS server and services
     Loaded: loaded (/lib/systemd/system/nfs-server.service; enabled; vendor preset: enabled)
     Active: active (exited) since Sun 2021-08-29 15:47:25 CEST; 10s ago
    Process: 7699 ExecStartPre=/usr/sbin/exportfs -r (code=exited, status=0/SUCCESS)
    Process: 7700 ExecStart=/usr/sbin/rpc.nfsd $RPCNFSDARGS (code=exited, status=0/SUCCESS)
   Main PID: 7700 (code=exited, status=0/SUCCESS)
        CPU: 8ms

août 29 15:47:24 server-nfs systemd[1]: Starting NFS server and services...
août 29 15:47:25 server-nfs systemd[1]: Finished NFS server and services.
```

Enfin, on consulte la liste des entrées exportées via NFS.

```
sudo exportfs
/home/exports   192.168.51.192/27
/home/exports   2001:678:3fc:1f5::/64
/home/exports/home
                192.168.51.192/27
/home/exports/home
                2001:678:3fc:1f5::/64
```

Cette dernière liste est identique à celle produite par la commande showmount côté client NFS.

Q66. Qu'est-ce qui distingue l'exportation d'une arborescence entre les versions 3 et 4 du protocole NFS ?

Rechercher dans les différences relatives à la notion de nommage dans les manipulations proposées dans les supports *Linux NFS-HOWTO* et *Nfsv4 configuration*.

Donner la signification du paramètre `fsid=0` dans la documentation relative à la version 4. Proposer une analogie avec le fonctionnement d'un serveur Web.

Au delà des évolutions du protocole, c'est la cohérence du système de nommage qui distingue la version 4 du système de fichiers réseau. Il s'agit de garantir qu'un objet (fichier ou répertoire) soit représenté de la même manière sur un serveur et sur ses clients.

Dans le contexte de ces travaux pratiques les répertoires utilisateurs doivent être référencés à partir d'une racine nommée `/ahome/`.

Du point de vue infrastructure, l'utilisation de cette référence de nommage unique présente un avantage non négligeable. En effet, les répertoires d'exportation tels qu'ils ont été définis dans le fichier `/etc/exports` donné ci-dessus désignent un espace de stockage physique.

La racine `/ahome/` désigne un espace de stockage logique. Ce schéma de nommage logique doit rester constant alors que les volumes de stockages physique peuvent migrer et se déplacer, être étendus, etc.

Les différences entre les manipulations proposées dans les supports *Linux NFS-HOWTO* et *Nfsv4 configuration* traduisent les différences de conception entre les deux générations du protocole NFS. On peut relever deux paramètres importants sur le serveur.

- L'option `fsid=0`, présente dans le fichier `/etc/exports/`, permet de définir une ***racine de montage*** tout comme on le verrait sur un serveur Web. Le paramètre de configuration `DocumentRoot /var/www` du serveur ***apache2*** désigne la racine à partir de laquelle les pages Web publiées sont référencées. Cette racine est indépendante de l'arborescence du système de fichier local du serveur.

- L'utilisation d'un montage local avec l'option `bind` de la commande mount permet de mettre en cohérence l'arborescence du serveur et de ses clients. Ainsi, le répertoire `/ahome/` présente les mêmes objets que l'on soit connecté sur le serveur ou sur un client. Le schéma de nommage est donc cohérent.

  Le montage local peut se faire manuellement sur le serveur avec la syntaxe suivante.

  ```
  sudo mkdir /ahome
  ```

  ```
  sudo mount --bind /home/exports/home /ahome
  ```

  Une fois la configuration validée, on peut intégrer ce montage local dans la configuration système pour que l'opération soit effectuée à chaque initialisation. Il faut alors éditer le fichier de configuration dédié aux montages des volumes locaux du système : `/etc/fstab`.

  Voici comment ajouter l'instruction de montage au fichier `/etc/fstab` du serveur NFS.

  ```
  echo "/home/exports/home     /ahome  none    defaults,bind    0    0" | \
  sudo tee -a /etc/fstab
  ```

  ```
  grep -v ^# /etc/fstab
  UUID=8362b3e6-d426-4f1b-93eb-e1efc22f60f4 /       ext4    errors=remount-ro 0    1
  UUID=f3e18b95-7430-4fea-ace5-7dd4cea6398a none    swap    sw    0       0
  /home/exports/home     /ahome  none    defaults,bind     0       0
  ```

Q67. Comment créer un compte utilisateur local baptisé `etu-nfs` avec un répertoire utilisateur situé sous la racine `/ahome` ?

Après consultation des pages de manuels de la commande adduser, on dispose des options de création de compte respectant le critère énoncé. L'option `--home` permet de désigner le répertoire utilisateur dans l'arborescence système.

```
sudo adduser --home /ahome/etu-nfs etu-nfs
```

```
id etu-nfs
uid=1001(etu-nfs) gid=1001(etu-nfs) groupes=1001(etu-nfs)
```

Les identifiants numériques `uid/gid` jouent un rôle important dans la suite des manipulations. Voir Section 6, « Gestion des droits sur le système de fichiers NFS ».

Q68. Créer un fichier texte ayant pour propriétaire l'utilisateur `etu-nfs` côté serveur et visualiser son contenu côté client.

Réaliser une capture et relever les numéros de ports caractéristiques de des transactions de montage. Est-il possible de retrouver le contenu du fichier texte dans les données de capture ?

Pour réaliser cette capture, il faut synchroniser les opérations entre les systèmes client et serveur. On commence par le lancement du l'analyseur réseau puis on visualise le contenu du fichier.

Côté serveur NFS, on créé le fichier texte puis on lance la capture réseau.

```
etu@server-nfs:~$ su - etu-nfs
Mot de passe :
etu-nfs@server-nfs:~$ echo "This file is mine" > textfile
etu-nfs@server-nfs:~$ exit
déconnexion
```

```
etu@server-nfs:~$ tshark -i enp0s1 -f "! port 22"
Capturing on 'enp0s1'
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 NFS 254 V4 Call GETATTR FH: 0x455db001
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 NFS 330 V4 Reply (Call In 3) GETATTR
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 86 883 → 2049 [ACK] Seq=169 Ack=245
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 NFS 262 V4 Call ACCESS FH: 0x455db001, [Check: RD LU
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 NFS 258 V4 Reply (Call In 6) ACCESS, [Allowed: RD LU
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 86 883 → 2049 [ACK] Seq=345 Ack=417
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 NFS 254 V4 Call GETATTR FH: 0x455db001
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 NFS 330 V4 Reply (Call In 9) GETATTR
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 86 883 → 2049 [ACK] Seq=513 Ack=661
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 NFS 278 V4 Call READDIR FH: 0x455db001
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 NFS 1174 V4 Reply (Call In 12) READDIR
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 86 883 → 2049 [ACK] Seq=705 Ack=1749
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 NFS 254 V4 Call GETATTR FH: 0x455db001
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 NFS 330 V4 Reply (Call In 15) GETATTR
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 86 883 → 2049 [ACK] Seq=873 Ack=1993
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 NFS 322 V4 Call OPEN DH: 0x6cceef4e/
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 NFS 442 V4 Reply (Call In 18) OPEN StateID: 0x5daa
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 86 883 → 2049 [ACK] Seq=1109 Ack=2349
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 NFS 270 V4 Call READ StateID: 0x7dca Offset: 0 Len: 1
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 NFS 214 V4 Reply (Call In 21) READ
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 86 883 → 2049 [ACK] Seq=1293 Ack=2477
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 NFS 262 V4 Call CLOSE StateID: 0x5daa
2001:678:3fc:1f5:baad:caff:fefe:3 → 2001:678:3fc:1f5:baad:caff:fefe:2 NFS 202 V4 Reply (Call In 24) CLOSE
2001:678:3fc:1f5:baad:caff:fefe:2 → 2001:678:3fc:1f5:baad:caff:fefe:3 TCP 86 883 → 2049 [ACK] Seq=1469 Ack=2593
```

Comme dans les opérations de capture réseau précédentes, il est préférable de stocker les résultats dans un fichier pour les exploiter ultérieurement avec une interface interactive qui permet d'isoler chaque champ de protocole.

Ici, on relève l'utilisation du protocole TCP en couche transport avec le port enregistré `2049/nfs`. Une analyse détaillée de l'appel de procédure `READ` montre que le contenu du fichier texte est bien visible.

5. Configuration du client NFS

Le rôle du client est d'intégrer un accès au système de fichiers d'un hôte distant dans son arborescence locale. On parle de «montage NFS». Dans un premier temps, on teste les opérations de montage manuel. Bien sûr, ces tests ne peuvent aboutir que si une arborescence à été exportée par un serveur.

Ensuite, on teste les opérations de montage automatisées ou ***automontage***. Si le serveur NFS n'est pas encore disponible au moment des tests de montage manuel, il faut préparer les fichiers de configuration du service d'automontage.

5.1. Opérations manuelles de (montage|démontage) NFS

Q69. Quelle est la commande qui permet de tester la disponibilité du service de montage NFS sur un hôte distant ?

Reprendre l'utilisation de la commande qui donne les listes des procédures distantes disponibles. Elle a été identifiée dans la section précédente.

Relativement aux résultats de la section précédente, la liste des services accessibles via RPC sur le serveur NFS s'est étoffée et le service de montage NFS apparaît clairement.

Voici un exemple de résultat utilisant l'adresse IP du serveur NFS.

```
rpcinfo -s fe80::baad:caff:fefe:3
   program version(s) netid(s)                         service     owner
    100000  2,3,4     local,udp,tcp,udp6,tcp6          portmapper  superuser
    100005  3,2,1     tcp6,udp6,tcp,udp                mountd      superuser
    100003  4,3       udp6,tcp6,udp,tcp                nfs         superuser
    100227  3         udp6,tcp6,udp,tcp                -           superuser
    100021  4,3,1     tcp6,udp6,tcp,udp                nlockmgr    superuser
```

Q70. Quelle est la commande qui permet d'identifier l'arborescence disponible à l'exportation depuis le serveur NFS ?

Rechercher dans la liste des commandes du paquet de service NFS commun au client et au serveur.

Dans la liste des commandes fournies avec le paquet `nfs-common`, on trouve un programme appelé showmount. Après consultation des pages de manuels, on relève l'option `-e` qui permet de consulter l'arborescence exportée par un serveur depuis un client. Voici un exemple d'exécution.

```
sudo showmount -e fe80::baad:caff:fefe:3
Export list for fe80::baad:caff:fefe:3:
/home/exports/home 2001:678:3fc:1f5::/64,192.168.51.192/27
/home/exports      2001:678:3fc:1f5::/64,192.168.51.192/27
```

Les résultats de la copie d'écran ci-dessus supposent que le serveur NFS ait déjà été configurer pour exporter le dossier `home`.

La commande showmount ne produit aucun résultat si le serveur NFS n'est pas configuré.

Q71. Quelle est la commande à utiliser pour les opérations de montage manuel ? À quel paquet appartient cette commande ? Cette commande est-elle exclusivement liée au protocole NFS ?

Après avoir consulté le support *Linux NFS-HOWTO*, interroger la base de données des paquets, rechercher dans le contenus des paquets et consulter les pages de manuels.

La documentation indique que c'est la commande mount qui nous intéresse. On effectue ensuite les recherches avec le gestionnaire de paquets.

```
apt search ^mount$
En train de trier... Fait
Recherche en texte intégral... Fait
mount/testing,now 2.37.2-1 amd64  [installé]
  tools for mounting and manipulating filesystems
```

```
dpkg -L mount | grep bin
/bin
/bin/mount
/bin/umount
/sbin
/sbin/losetup
/sbin/swapoff
/sbin/swapon
```

La commande appartient au paquet du même nom. La consultation des pages de manuels `$ man mount` montre que cette commande n'est pas réservée au seul protocole NFS mais à l'ensemble des opérations de montage pour tous les systèmes de fichiers utilisables.

Q72. Créer le répertoire `/ahome` destiné à «recevoir» le contenu répertoires utilisateurs exportés depuis le serveur NFS. Quelle est la syntaxe de la commande permettant de ***monter*** le répertoire exporté par le serveur NFS sur ce nouveau répertoire ?

Rechercher dans le support *Linux NFS-HOWTO*.

Exemple avec l'adresse IPv6 du serveur NFS.

```
sudo mkdir /ahome
```

```
sudo mount [2001:678:3fc:1f5:baad:caff:fefe:3]:/home /ahome
```

```
mount | grep nfs
[2001:678:3fc:1f5:baad:caff:fefe:3]:/home on /ahome type nfs4 \
 (rw,relatime,vers=4.2,rsize=131072,wsize=131072,namlen=255,hard,proto=tcp6,
 timeo=600,retrans=2,sec=sys,clientaddr=2001:678:3fc:1f5:baad:caff:fefe:2,
 local_lock=none,addr=2001:678:3fc:1f5:baad:caff:fefe:3)
```

Exemple avec l'adresse IPv4 du serveur NFS.

```
sudo mkdir /ahome
```

```
sudo mount 192.168.51.195:/home /ahome
```

```
mount | grep nfs
192.168.51.195:/home on /ahome type nfs4 \
 (rw,relatime,vers=4.2,rsize=131072,wsize=131072,namlen=255,hard,proto=tcp,
 timeo=600,retrans=2,sec=sys,clientaddr=192.168.51.194,
 local_lock=none,addr=192.168.51.195)
```

Q73. Réaliser une capture lors de l'exécution de la commande ls -lAh /ahome et relever les numéros de ports caractéristiques de ces transactions. Est-il possible de retrouver les informations échangées dans les données de capture ?

La marche à suivre est identique à celle de la même question côté serveur NFS.

1. On lance la capture de trafic côté serveur NFS.

```
tshark -i enp0s1 -f "! port 22" -w /var/tmp/ls-nfs.pcap
```

2. On exécute la commande ls -lAh /ahome côté client NFS.

3. Retour côté serveur pour exploiter les résultats.

L'analyse montre que le protocole NFS en version 4 utilise bien le mode `COMPOUND` de traitement par lot des appels de procédure distants RPC. On ne relève dans cette capture que les métadonnées système sur les attributs et les permissions relatives à l'arborescence lue.

Si on reprend la même démarche avec la commande cat d'un fichier texte par exemple, le contenu de ce fichier apparaît en clair dans la capture de trafic.

Q74. Quelles ***seraient*** les opérations à effectuer pour configurer le système et rendre un montage NFS statique permanent ?

Rechercher le fichier de configuration système responsable des montages statiques des partitions.

Il est inutile de modifier les fichiers de configuration du système sachant que l'on change de méthode de montage dans la section suivante.

Il faudrait éditer le fichier `/etc/fstab` pour effectuer un montage statique à chaque initialisation du système. On pourrait par exemple insérer une ligne du type suivant à la fin du fichier.

- Avec le protocole IPv4 :

```
192.168.51.195:/home   /ahome   nfs4    0   0
```

- Avec le protocole IPv6 :

```
[2001:678:3fc:1f5:baad:caff:fefe:3]:/home   /ahome   nfs4    0   0
```

Q75. Quelle est la commande à utiliser pour ***démonter*** le dossier `/ahome` ?

Rechercher cette commande dans la liste des outils forunis avec le paquet `mount`.

C'est la commande umount qu'il faut utiliser pour «détacher» un dispositif de stockage du système de fichiers. Dans le cas de cette section, la syntaxe est la suivante.

```
sudo umount /ahome
```

5.2. Opérations automatisées de (montage|démontage) NFS

Dans cette section, on reprend le processus de montage précédent en utilisant le service d'automontage. L'objectif étant de rendre les opérations d'accès au système de fichiers réseau totalement transparentes pour l'utilisateur, le recours au montage manuel doit être évité le plus possible.

Il existe plusieurs implémentations libres pour le service d'automontage. On se limite ici au logiciel lié au noyau Linux.

Avertissement

Les montages manuels et le service d'automontage ne font pas bon ménage ! Il faut absolument démonter tous les systèmes de fichiers NFS avant d'aborder cette partie.

Q76. Quel est le paquet qui contient les outils nécessaires au fonctionnement de l'automontage ?

Rechercher le mot clé automount dans les descriptions du gestionnaire de paquets.

```
aptitude search "?description(automount)"
p   afuse                           - automounting file system implemented in user-space
p   autodir                         - Automatically creates home and group directories
p   autofs                          - kernel-based automounter for Linux
p   autofs-hesiod                   - Hesiod map support for autofs
p   autofs-ldap                     - LDAP map support for autofs
p   fusiondirectory-plugin-autofs   - autofs plugin for FusionDirectory
p   libnss-cache                    - NSS module for using nsscache-generated fi
p   libunix-configfile-perl         - Perl interface to various Unix configurati
p   nsscache                        - asynchronously synchronise local NSS databases
p   pmount                          - mount removable devices as normal user
i   systemd                         - system and service manager
i   systemd-sysv                    - system and service manager - SysV links
p   udevil                          - Alternative storage media interface
p   udiskie                         - automounter for removable media for Python
p   vfu                             - Versatile text-based file-manager
```

Dans le contexte de ces manipulations, c'est le paquet `autofs` qui nous intéresse.

```
sudo apt install autofs
```

Q77. Comment créer un compte utilisateur local baptisé `etu-nfs` avec un répertoire utilisateur situé sous la racine `/ahome` dont les fichiers et répertoires sont placés sur le serveur NFS ?

Après consultation des pages de manuels de la commande adduser, on dispose des options de création de compte respectant les deux critères énoncés. L'option `--home` permet de désigner le répertoire utilisateur dans l'arborescence système et l'option `--no-create-home` évite la création de ce répertoire sur le système local.

```
sudo adduser --no-create-home --home /ahome/etu-nfs etu-nfs
Attention ! Impossible d'accéder au répertoire personnel que vous avez indiqué (/ahome/etu-nfs) : No such file or directory
Ajout de l'utilisateur « etu-nfs » ...
Ajout du nouveau groupe « etu-nfs » (1001) ...
Ajout du nouvel utilisateur « etu-nfs » (1001) avec le groupe « etu-nfs » ...
Le répertoire personnel « /ahome/etu-nfs » n'a pas été créé.
Nouveau mot de passe :
Retapez le nouveau mot de passe :
passwd: password updated successfully
Changing the user information for etu-nfs
Enter the new value, or press ENTER for the default
        Full Name []: Etudiant NFS
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Cette information est-elle correcte ? [O/n]
```

```
id etu-nfs
uid=1001(etu-nfs) gid=1001(etu-nfs) groupes=1001(etu-nfs)
```

Les identifiants numériques `uid/gid` jouent un rôle important dans la suite des manipulations. Voir Section 6, « Gestion des droits sur le système de fichiers NFS ».

Q78. Quels sont les fichiers de configuration du service d'automontage à éditer ou créer pour que l'utilisateur `etu-nfs` ait accès à ses données personnelles ?

Utiliser les fichiers exemples fournis avec le paquet, les pages de manuels associées et créer un fichier spécifique pour la gestion des comptes utilisateurs.

La liste des fichiers du paquet `autofs` montre qu'il existe une page de manuel consacrée au fichier principal de configuration du service : `/etc/auto.master`. Ces informations permettent de configurer un point de montage au dessous duquel doivent se trouver les répertoires utilisateurs. Ces derniers utilisent un fichier de configuration propre : `/etc/auto.home`.

1. On définit la racine de montage `/ahome` dans le fichier de configuration principal `/etc/auto.master`. Cette racine de montage pointe vers le fichier de configuration dédié au montage automatique des répertoires des utilisateurs.

   Après analyse des commentaires présents dans le fichier `/etc/auto.master`, on créé un fichier spécifique à notre contexte dans le dossier `/etc/auto.master.d/` avec le suffixe `.autofs`.

   ```
   echo "/ahome  /etc/auto.home" | \
   sudo tee -a /etc/auto.master.d/ahome.autofs
   ```

2. On créé le fichier `/etc/auto.home` qui utilise une syntaxe particulière pour que le montage du système de fichiers du serveur soit générique et indépendant du nombre des comptes utilisateurs.

```
echo "*    -fstype=nfs4    [2001:678:3fc:1f5:baad:caff:fefe:3]:/home/&" | \
sudo tee -a /etc/auto.home
```

- Le premier paramètre est le symbole * qui se substitue au nom d'utilisateur : `etu-nfs` dans notre exemple.
- Le deuxième paramètre `-fstype=nfs4` correspond à une option de montage qui privilégie la version 4 du protocole NFS. Le jeu des options de montage est le même que pour un montage statique.
- Le troisième paramètre est l'adresse IPv4 ou IPv6 du serveur. Comme on ne dispose pas d'un service DNS à ce stade de la progression des travaux pratiques, on utilise directement les adresses IP.
- Le répertoire `/home/` correspond à la configuration de l'exportation NFS sur le serveur. Le répertoire `/home/` est situé sous la racine d'exportation qui est uniquement connue du serveur.
- Le symbole & indique la répétition du premier paramètre : le nom d'utilisateur.

3. Une fois les fichiers de configuration en place, il ne faut pas oublier de redémarrer le service et de contrôler son bon fonctionnement.

```
sudo systemctl restart autofs
```

```
systemctl status autofs
# autofs.service - Automounts filesystems on demand
     Loaded: loaded (/lib/systemd/system/autofs.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2021-08-29 09:42:16 CEST; 51s ago
       Docs: man:autofs(8)
    Process: 8027 ExecStart=/usr/sbin/automount $OPTIONS --pid-file /var/run/autofs.pid (code=exited, status=0/SUCCESS
   Main PID: 8028 (automount)
      Tasks: 4 (limit: 1131)
     Memory: 1.0M
        CPU: 29ms
     CGroup: /system.slice/autofs.service
             └─8028 /usr/sbin/automount --pid-file /var/run/autofs.pid

août 29 09:42:16 client-nfs systemd[1]: Starting Automounts filesystems on demand...
août 29 09:42:16 client-nfs systemd[1]: Started Automounts filesystems on demand.
```

Q79. Quelles sont les conditions à respecter sur le client et le serveur NFS pour que l'utilisateur `etu-nfs` ait la capacité à écrire dans son répertoire personnel ?

Rechercher les attributs d'un compte utilisateur qui correspondent aux propriétés des objets d'un système de fichiers au sens général.

Les identifiants numériques `uid/gid` doivent nécessairement être identiques sur le client et le serveur NFS. Toute la gestion des droits sur le système de fichiers est conditionnée par ces valeurs.

Q80. Comment prendre l'identité de l'utilisateur `etu-nfs` pour tester la validité du montage ?

Cette validation suppose que l'utilisateur puisse atteindre son répertoire et que l'on visualise l'automontage avec les commandes mount et df.

C'est la commande su qui permet de «changer d'identité» sur le système. On l'utilise donc pour prendre l'identité de l'utilisateur dont le répertoire est situé sur le serveur NFS. Pour que l'opération de montage automatique ait lieu, il suffit de se placer dans ce répertoire.

```
etu@client-nfs:~$ su - etu-nfs
etu-nfs@client-nfs:~$ pwd
/ahome/etu-nfs
etu-nfs@:client-nfs~$ df -HT
Sys. de fichiers                                  Type     Taille Utilisé Dispo Uti% Monté sur
udev                                              devtmpfs   495M       0  495M   0% /dev
tmpfs                                             tmpfs      103M    680k  102M   1% /run
/dev/vda1                                         ext4        72G    2,4G   66G   4% /
tmpfs                                             tmpfs      512M       0  512M   0% /dev/shm
tmpfs                                             tmpfs      5,3M       0  5,3M   0% /run/lock
tmpfs                                             tmpfs      103M       0  103M   0% /run/user/1000
[2001:678:3fc:1f5:baad:caff:fefe:3]:/home/etu-nfs nfs4        72G    2,4G   66G   4% /ahome/etu-nfs
```

```
etu-nfs@client-nfs:~$ mount | grep nfs
[2001:678:3fc:1f5:baad:caff:fefe:3]:/home/etu-nfs on /ahome/etu-nfs type nfs4
 (rw,relatime,vers=4.2,rsize=131072,wsize=131072,namlen=255,hard,proto=tcp6,
 timeo=600,retrans=2,sec=sys,clientaddr=2001:678:3fc:1f5:baad:caff:fefe:2,
 local_lock=none,addr=2001:678:3fc:1f5:baad:caff:fefe:3)
```

Bien sûr, ces manipulations ne sont possibles que si la configuration du serveur est effective.

Q81. Réaliser une capture réseau lors de l'exécution des commandes et relever les numéros de ports caractéristiques de ces transactions. Est-il possible de retrouver les informations échangées dans les données de capture ?

La marche à suivre est identique à celle de la même question côté serveur NFS.

6. Gestion des droits sur le système de fichiers NFS

Le contrôle les droits sur les objets de l'arborescence exportée par le serveur NFS est limité au masque de permissions de ces objets. Il est donc important de faire correspondre les identifiants `uid` et `gid` entre le client et le serveur.

Les manipulations suivantes sont à réaliser en «concertation» entre les administrateurs des postes client et serveur. Le compte utilisateur `etu-nfs` doit avoir été créé sur le serveur et sur le client.

> Note
>
> Ces manipulations se font sans système de gestion centralisé de l'authentification. L'utilisation d'un annuaire LDAP pour fournir une base de comptes utilisateurs fait l'objet d'un support de travaux pratiques qui vient après celui-ci. Ce support se concentre sur le volet système de fichiers réseau.

Q82. Quelles sont les valeurs numériques des identifiants `uid` et `gid` du compte utilisateur `etu-nfs` sur le client et sur le serveur NFS ?

Si les valeurs différent entre le client et le serveur, il faut détruire ces comptes utilisateurs et reprendre les options de la commande adduser pour fournir ces valeurs de façon explicite.

L'extrait du résultat de l'instruction `$ sudo adduser --help` ci-dessous montre les options utiles.

```
adduser [--home DIR] [--shell SHELL] [--no-create-home] [--uid ID]
[--firstuid ID] [--lastuid ID] [--gecos GECOS] [--ingroup GROUP | --gid ID]
[--disabled-password] [--disabled-login] USER
   Ajoute un utilisateur normal
```

Reprendre la question sur la création d'un compte utilisateur local dont le répertoire est situé sur le serveur NFS.

Q83. Sur quel poste peut on créer des fichiers et des répertoires avec des masques de permissions ayant d'autres valeurs `uid` et `gid` que celles de l'utilisateur `etu-nfs` ? Quelles sont les options des commandes chmod et chown à utiliser pour réaliser ces opérations ?

Utiliser les pages de manuels des commandes.

C'est sur le serveur que le super utilisateur a la possibilité de créer n'importe quel objet avec n'importe quel propriétaire dans la mesure où le système de fichiers est local et non réseau.

```
etu@server-nfs:~$ sudo touch /ahome/etu-nfs/ThisOneIsMine
etu@server-nfs:~$ sudo chown etu-nfs.etu-nfs /ahome/etu-nfs/ThisOneIsMine
etu@server-nfs:~$ sudo touch /ahome/etu-nfs/ThisOneIs-NOT-Mine
etu@server-nfs:~$ sudo chown 2000.2000 /ahome/etu-nfs/ThisOneIs-NOT-Mine
etu@server-nfs:~$ sudo ls -lh /ahome/etu-nfs/
total 4,0K
-rw-r--r-- 1 etu-nfs etu-nfs 18 29 août  16:15 textfile
-rw-r--r-- 1 etu-nfs etu-nfs  0 29 août  18:32 ThisOneIsMine
-rw-r--r-- 1    2000    2000  0 29 août  18:33 ThisOneIs-NOT-Mine
```

Côté client, les objets créés sont biens visibles et la vue réseau du système de fichiers NFS passe par une correspondance des propriétaires.

```
etu-nfs@client-nfs:~$ id
uid=1001(etu-nfs) gid=1001(etu-nfs) groupes=1001(etu-nfs)
etu-nfs@client-nfs:~$ ls -lh
total 4,0K
-rw-r--r-- 1 etu-nfs etu-nfs 18 29 août  16:15 textfile
-rw-r--r-- 1 etu-nfs etu-nfs  0 29 août  18:32 ThisOneIsMine
-rw-r--r-- 1    2000    2000  0 29 août  18:33 ThisOneIs-NOT-Mine
```

Côté client NFS, les valeurs des identifiants `uid` et `gid` sont correctement restitués et l'utilisateur n'a que le droit de lecture sur le fichier `ThisOneIs-NOT-Mine`.

Q84. Quel est le service qui assure la conformité des identifiants entre serveur et client NFS ?

Reprendre la liste des service RPC actifs sur les deux systèmes.

Le démon `rpc.idmapd` est fourni avec le paquet `nfs-common`.

7. Documents de référence

Systèmes de fichiers réseau : NFS & CIFS

Systèmes de fichiers réseau : présentation des modes de fonctionnement des systèmes de fichiers réseau NFS & CIFS.

Linux NFS-HOWTO

Linux NFS-HOWTO : documentation historique complète sur la configuration d'un serveur et d'un client NFS jusqu'à la version 3 inclue.

Nfsv4 configuration

Nfsv4 configuration : traduction française extraite des pages du projet CITI de l'université du Michigan.

Introduction aux annuaires LDAP avec OpenLDAP

https://www.inetdoc.net

Résumé

Dans ce support de travaux pratiques, on explore le service d'annuaire LDAP. On présente succinctement les éléments constitutifs d'un annuaire puis on étudie la configuration d'un service d'annuaire basé sur le logiciel OpenLDAP. Ensuite, on étudie la configuration de l'accès aux entrées de l'annuaire depuis un poste client. Les informations délivrées par l'annuaire sont les propriétés de comptes utilisateurs stockées dans la classe d'objet `posixAccount`.

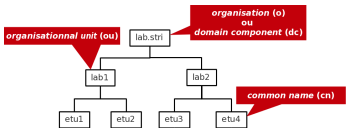


Table des matières

1. Principes d'un annuaire LDAP

Dans l'histoire des systèmes Unix, les services de ***nommage*** ont connu de nombreuses évolutions avec le développement de l'Internet et des volumes d'informations à partager.

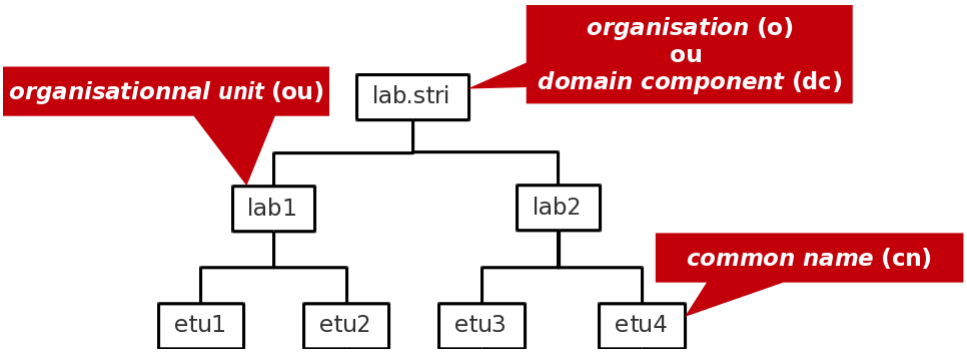
Au début des années 80, un premier service baptisé *Network Information Service* (NIS) a vu le jour. Ce service est une méthode de distribution de la base de données des utilisateurs, de fichiers de configuration, d'authentification et d'autres données entre les hôtes d'un réseau local. Le logiciel NIS développé par Sun Microsystems™ fonctionne sur le mode Client/Serveur à partir d'une base de données «à plat» (*flat bindery base*). Son utilisation est étudiée dans le support de travaux pratiques *Introduction au service NIS*. Avec un service NIS, il n'est pas possible de constituer des groupes logiques ayant des attributs propres. Cette limitation est rapidement devenue critique avec l'augmentation du nombres des utilisateurs et des clients.

D'autres services plus complets tels que NIS+ ou *kerberos* qui n'assure que la partie authentification ont été développés par la suite. Depuis quelques années, les annuaires LDAP ou *Lightweight Directory Access Protocol* se sont imposés comme étant l'outil d'échange universel des paramètres utilisateurs.

Pour définir ce qu'est le service LDAP, on peut retenir les caractéristiques suivantes.

- Un service de publication d'annuaire
- Un protocole d'accès aux annuaires de type X.500 ou *Lightweight Directory Access Protocol*
- Un dépôt de données basées sur des attributs ou un «genre» de base de données
- Un logiciel optimisé pour les recherches avancées et les lectures
- Une implémentation client/serveur
- Un mécanisme extensible de schémas de description de classes d'objets

Les entrées (*Directory Service Entry*) d'un annuaire LDAP sont distribuées suivant une arborescence (*Directory Information Tree*) hiérarchisée que l'on peut voir comme un système de fichiers avec ses répertoires et ses fichiers. Au sommet de l'arborescence on trouve un nom de racine (*Domain Component*) ou suffixe.



Arborescence LDAP élémentaire - vue complète

L'adresse d'une entrée de l'annuaire LDAP est appelée : ***distinguished name*** ou dn. En reprenant l'exemple d'arborescence ci-dessus, les adresses des différentes entrées sont notées comme suit.

- `dn: dc=lab,dc=stri`
- `dn: ou=lab1,dc=lab,dc=stri`
  `dn: ou=lab2,dc=lab,dc=stri`
- `dn: cn=etu1,ou=lab1,dc=lab,dc=stri`
  `dn: cn=etu2,ou=lab1,dc=lab,dc=stri`
  `dn: cn=etu3,ou=lab2,dc=lab,dc=stri`
  `dn: cn=etu4,ou=lab2,dc=lab,dc=stri`

L'adresse de chaque entrée appartient à une classe d'objet (*ObjectClass*) spécifiée dans un schéma (*schema*). En reprenant les mêmes exemples d'entrées, on peut associer les classes d'objets correspondantes.

entry	***objectclass***

`o: lab.stri`
`dc: lab`
`dc: stri`
	`organisation`
	`dcObject`
	`dcObject`

`ou: lab1`	`organisationalUnit`
`cn: etu1` `sn: etu1`	`inetOrgPerson`

Un schéma peut être vu comme un ensemble de règles qui décrivent la nature des données stockées. C'est un outil qui aide à maintenir la cohérence, la qualité et qui évite la duplication des données dans l'annuaire. Les attributs des classes d'objets déterminent les règles qui doivent être appliquées à une entrée. Un schéma contient les éléments suivants.

- Les attributs requis
- Les attributs autorisés
- Les règles de comparaison des attributs
- Les valeurs limites qu'un attribut peut recevoir
- Les restrictions sur les informations qui peuvent être enregistrées

2. Configuration du serveur LDAP

Avant d'aborder la configuration du service LDAP, il faut passer par les étapes rituelles de sélection et d'installation des paquets contenant les outils logiciels du service. Ensuite, il faut identifier les processus, les numéros de ports ouverts et les fichiers de configuration à éditer.

2.1. Installation du serveur LDAP

Q85. Quels sont les paquets Debian relatifs au service LDAP ?

Interroger la base de données des paquets pour obtenir les informations demandées.

Dans la requête ci-dessous, on privilégie la recherche dans les champs de description des paquets.

```
apt search ^OpenLDAP
```

```
En train de trier... Fait
Recherche en texte intégral... Fait
ldap-utils/testing 2.5.13+dfsg-5 amd64
  OpenLDAP utilities

libldap-2.5-0/testing,now 2.5.13+dfsg-5 amd64  [installé, automatique]
  Bibliothèques OpenLDAP

libldap-common/testing,now 2.5.13+dfsg-5 all  [installé, automatique]
  fichiers communs OpenLDAP pour les bibliothèques

libldap-dev/testing 2.5.13+dfsg-5 amd64
  bibliothèques de développement pour OpenLDAP

ruby-ldap/testing 0.9.20-2+b5 amd64
  OpenLDAP library binding for Ruby

slapd/testing 2.5.13+dfsg-5 amd64
  OpenLDAP server (slapd)
```

Q86. Quels sont les paquets Debian à installer pour mettre en œuvre un serveur LDAP ?

Dans liste obtenue en réponse à la question précédente, rechercher les paquets relatifs aux utilitaires et au serveur.

Dans la liste ci-dessus, on retient deux paquets : `ldap-utils` et `slapd`.

```
sudo apt -y install slapd ldap-utils
```

```
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  libltdl7 libodbc2
Paquets suggérés :
  libsasl2-modules-gssapi-mit | libsasl2-modules-gssapi-heimdal odbc-postgresql tdsodbc
Les NOUVEAUX paquets suivants seront installés :
  ldap-utils libltdl7 libodbc2 slapd
0 mis à jour, 4 nouvellement installés, 0 à enlever et 0 non mis à jour.
```

Lors de l'installation, deux écrans `debconf` demandent la saisie du mot de passe administrateur du service LDAP.

Q87. Comment identifier le ou les processus correspondant au service installé ?

Utiliser une commande d'affichage de la liste des processus actifs sur le système pour identifier le démon correspondant au serveur LDAP.

```
ps aux | grep  l[d]ap
```

```
openldap    1699  0.0  1.0 1159776 10540 ?       Ssl  18:22   0:00
  /usr/sbin/slapd -h ldap:/// ldapi:/// -g openldap -u openldap -F /etc/ldap/slapd.d
```

À partir de ces informations, on identifie le démon serveur `slapd`, le compte utilisateur et le groupe système propriétaires du processus (`openldap`) et enfin le répertoire contenant les fichiers de configuration `/etc/ldap/slapd.d`.

Q88. Comment identifier le ou les numéros de ports ouverts par le service installé ?

Utiliser une commande d'affichage de la liste des ports ouverts sur le système.

Voici deux exemples usuels.

```
sudo lsof -i | grep l[d]ap
```

```
slapd   1699 openldap    8u  IPv4  19101      0t0  TCP *:ldap (LISTEN)
slapd   1699 openldap    9u  IPv6  19102      0t0  TCP *:ldap (LISTEN)
```

```
ss -tau | grep l[d]ap
```

```
tcp   LISTEN 0      2048        0.0.0.0:ldap            0.0.0.0:*
tcp   LISTEN 0      2048           [::]:ldap               [::]:*
```

Les numéros de port enregistrés pour le service LDAP sont disponibles dans le fichier `/etc/services`.

```
grep ldap /etc/services
```

```
ldap            389/tcp         # Lightweight Directory Access Protocol
ldap            389/udp
ldaps           636/tcp         # LDAP over SSL
ldaps           636/udp
```

Relativement au indications données par les commandes lsof et ss, c'est le numéro de port 389 qui est ouvert en écoute lors de l'installation du paquet `slapd`.

Par défaut l'accès TLS au service n'est pas activé.

2.2. Analyse de la configuration du service LDAP

Les versions actuelles du logiciel ***OpenLDAP*** utilisent un mode de configuration basé sur un ***Directory Information Tree*** ou DIT propre. Cette arborescence de configuration est pointée par le nom `cn=config`. Elle est utilisée pour configurer dynamiquement le démon `slapd`, modifier les définitions de schéma, les index, les listes de contrôle d'accès ACLs, etc. Ce mode de configuration présente un avantage déterminant lorsque l'on exploite des annuaires volumineux : toutes les opérations se font sans interruption de service.

Les documents fournis avec le paquet `slapd` contiennent des informations indispensables à l'analyse du fonctionnement du service.

Q89. Quel est le mode de gestion de la configuration du service du paquet de la distribution Debian GNU/Linux ?

Consulter les fichiers de documentation fournis avec le paquet `slapd`.

Les documents relatifs au paquet `slapd` sont situés dans le répertoire `/usr/share/doc/slapd/`. Le fichier `README.Debian.gz` contient un exemple d'instruction de consultation de la configuration du service.

```
sudo ldapsearch -Y EXTERNAL -H ldapi:/// -b "cn=config"
```

Q90. Quel est le gestionnaire de base de données (***backend***) proposé dans l'annuaire de configuration ?

Reprendre la commande préconisée en réponse à la question précédente en utilisant le type de base de donnée comme filtre.

```
sudo ldapsearch -Y EXTERNAL -H ldapi:/// -b "cn=config" \
  olcDatabase={1}mdb
```

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
# extended LDIF
#
# LDAPv3
# base <cn=config> with scope subtree
# filter: olcDatabase={1}mdb
# requesting: ALL
#

# {1}mdb, config
dn: olcDatabase={1}mdb,cn=config
objectClass: olcDatabaseConfig
objectClass: olcMdbConfig
olcDatabase: {1}mdb
olcDbDirectory: /var/lib/ldap
olcSuffix: dc=nodomain
olcAccess: {0}to attrs=userPassword by self write by anonymous auth by * none
olcAccess: {1}to attrs=shadowLastChange by self write by * read
olcAccess: {2}to * by * read
olcLastMod: TRUE
olcRootDN: cn=admin,dc=nodomain
olcRootPW: {SSHA}y3201Tkxe0HgfQOhLxiVJ3wwI8+dnQwK
olcDbCheckpoint: 512 30
olcDbIndex: objectClass eq
olcDbIndex: cn,uid eq
olcDbIndex: uidNumber,gidNumber eq
olcDbIndex: member,memberUid eq
olcDbMaxSize: 1073741824

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Par définition, un annuaire LDAP est une base de données optimisée en lecture. Du point de vue implémentation, les entrées sont stockées sous forme «binaire» et indexées à l'aide d'un gestionnaire de base de données. Le gestionnaire d'arrière plan proposé par défaut est `mdb`. Il s'agit d'une variante actualisée du gestionnaire ***Berkeley DB transactional backend***.

Q91. Comment identifier le nom de l'annuaire fourni par défaut avec le paquet `slapd` ?

Rechercher la clé `olcSuffix` dans la configuration de l'annuaire.

```
sudo ldapsearch -LLL -Y EXTERNAL -H ldapi:/// -b "cn=config" \
  olcSuffix | grep ^olcSuffix
```

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
olcSuffix: dc=nodomain
```

Q92. Quels sont les ***schemas*** actifs avec la configuration courante du paquet `slapd` ?

Rechercher la clé `olcSchemaConfig` dans la configuration de l'annuaire.

```
sudo ldapsearch -LLL -Y EXTERNAL -H ldapi:/// -b "cn=config" \
  olcSchemaConfig | grep ^cn
```

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
cn: config
cn: module{0}
cn: schema
cn: {0}core
cn: {1}cosine
cn: {2}nis
cn: {3}inetorgperson
```

Q93. Où sont stockées les bases définies par défaut lors de l'installation du paquet `slapd` ?

Rechercher la clé `olcDbDirectory` dans la configuration de l'annuaire.

```
sudo ldapsearch -Y EXTERNAL -H ldapi:/// -b "cn=config" \
  olcDbDirectory | grep ^olcDbDirectory
```

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
olcDbDirectory: /var/lib/ldap
```

C'est dans le répertoire `/var/lib/ldap` que sont stockées les fichiers des bases ***Berkeley DB***.

```
ls -lAh /var/lib/ldap/
```

```
total 40K
-rw------- 1 openldap openldap  36K  2 sept. 18:22 data.mdb
-rw------- 1 openldap openldap 8,0K  2 sept. 18:22 lock.mdb
```

2.3. Réinitialisation de la base de l'annuaire LDAP

L'installation du paquet `slapd` implique l'installation d'un annuaire minimal avec une base associée. Ce mode opératoire est nécessaire, ne serait-ce que pour accéder à la configuration du service et tester la validité de l'installation. Après avoir traité les questions ci-dessus, on sait que l'installation est fonctionnelle. On peut donc passer à l'initialisation de notre propre annuaire.

Note

Les manipulations proposées dans cette section permettent de reprendre à zéro la configuration d'un annuaire LDAP. Il peut être utile de revenir à cette étape en cas de «doute» sur l'intégrité de l'annuaire lors du traitement des questions des sections suivantes.

Q94. Comment arrêter le service LDAP ?

Utiliser les scripts fournis avec le gestionnaire de lancement des processus système.

Chaque processus système dispose d'un script de gestion de son lancement, arrêt (et|ou) redémarrage. Avec le gestionnaire systemd, il faut faire une recherche dans la liste des services. Une fois le service identifié, on l'arrête avec la commande systemctl.

```
systemctl status slapd
```

```
# slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Access Protocol)
     Loaded: loaded (/etc/init.d/slapd; generated)
    Drop-In: /usr/lib/systemd/system/slapd.service.d
             └─slapd-remain-after-exit.conf
     Active: active (running) since Sat 2023-09-02 18:22:14 CEST; 21min ago
       Docs: man:systemd-sysv-generator(8)
    Process: 1693 ExecStart=/etc/init.d/slapd start (code=exited, status=0/SUCCESS)
      Tasks: 4 (limit: 1084)
     Memory: 7.4M
        CPU: 59ms
     CGroup: /system.slice/slapd.service
             └─1699 /usr/sbin/slapd -h "ldap:/// ldapi:///" -g openldap -u openldap -F /etc/ldap/slapd.d

sept. 02 18:22:14 ldap-server systemd[1]: Starting slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory A
sept. 02 18:22:14 ldap-server slapd[1698]: @(#) $OpenLDAP: slapd 2.5.13+dfsg-5 (Feb  8 2023 01:56:12) $
                                                   Debian OpenLDAP Maintainers <pkg-openldap-devel@lists.alioth.debian.org>
sept. 02 18:22:14 ldap-server slapd[1699]: slapd starting
sept. 02 18:22:14 ldap-server slapd[1693]: Starting OpenLDAP: slapd.
sept. 02 18:22:14 ldap-server systemd[1]: Started slapd.service - LSB: OpenLDAP standalone server (Lightweight Directory Ac
```

Instruction d'arrêt du service :

```
sudo systemctl stop slapd
```

On peut exécuter à nouveau la commande systemctl status slapd pour confirmer que le service est bien stoppé et inactif.

Q95. Quels sont les éléments à supprimer pour pouvoir installer une nouvelle configuration et une nouvelle base LDAP ?

Utiliser le résultat de la question sur la localisation des bases et la documentation fournie avec le paquet `slapd`.

À partir des réponses aux questions ci-dessus, on sait que c'est le répertoire `/var/lib/ldap/` qui contient les bases de données du service. La lecture du fichier de documentation du paquet avec la commande `zless /usr/share/doc/slapd/README.Debian.gz` indique que les fichiers de configuration sont situés dans le répertoire `/etc/ldap/slapd.d/`.

On supprime donc tous ces fichiers et répertoires.

```
sudo rm -rf /var/lib/ldap/* /etc/ldap/slapd.d
```

Q96. Comment reprendre à zéro la configuration du paquet `slapd` ?

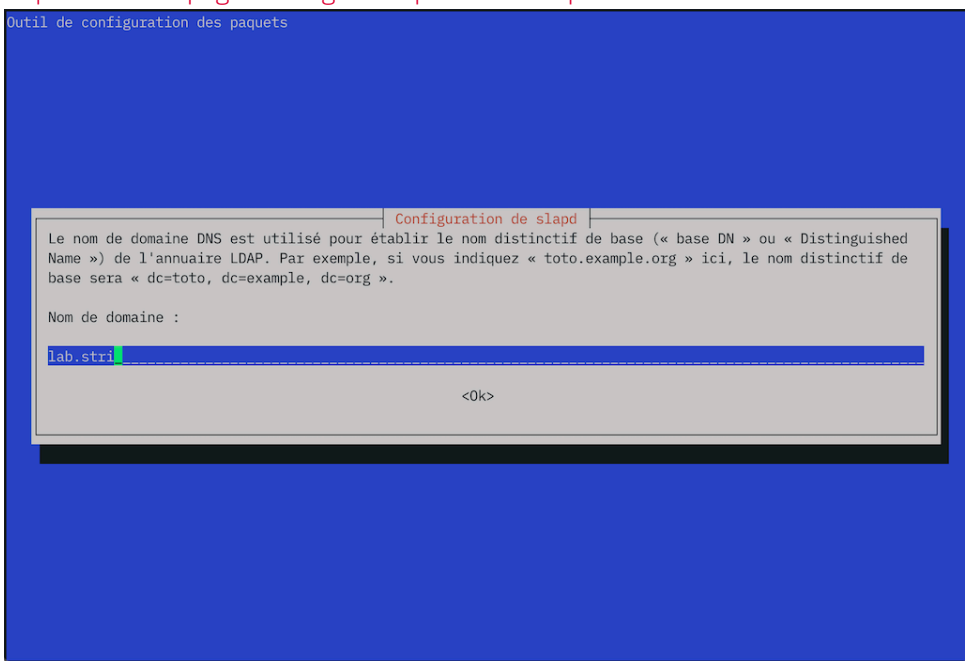
Utiliser l'outil du gestionnaire de paquets ***Debian GNU/Linux*** qui permet la modification des paramètres de configuration d'un service à l'aide de menus `debconf`.

C'est la commande dpkg-reconfigure qui sert à réviser les paramètres de configuration d'un paquet. Voici une copie des écrans proposés avec le paquet `slapd`.

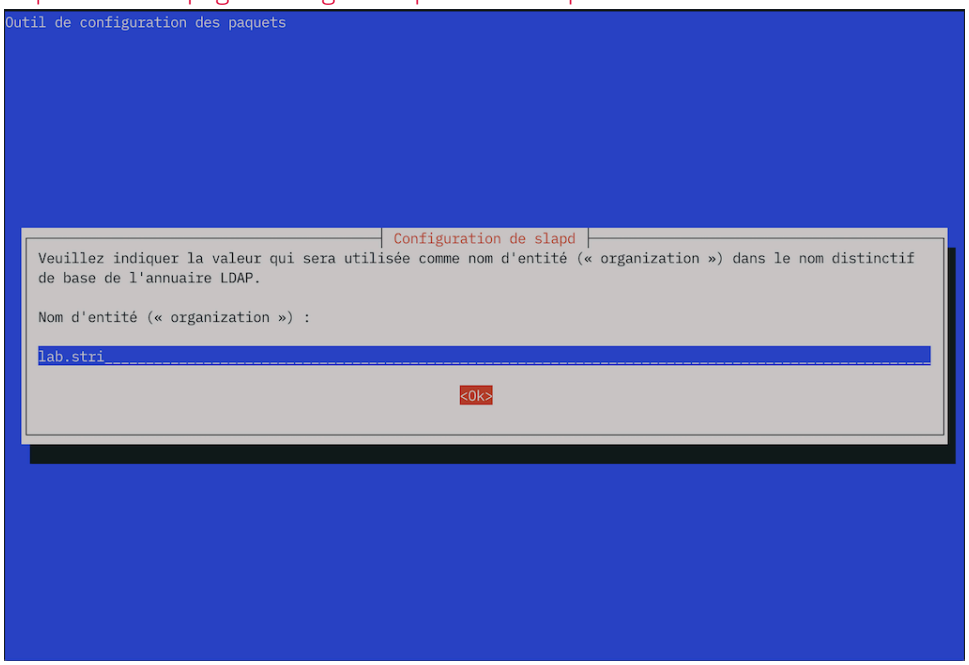
```
sudo dpkg-reconfigure slapd
```

```
  Creating initial configuration... done.
  Creating LDAP directory... done.
```

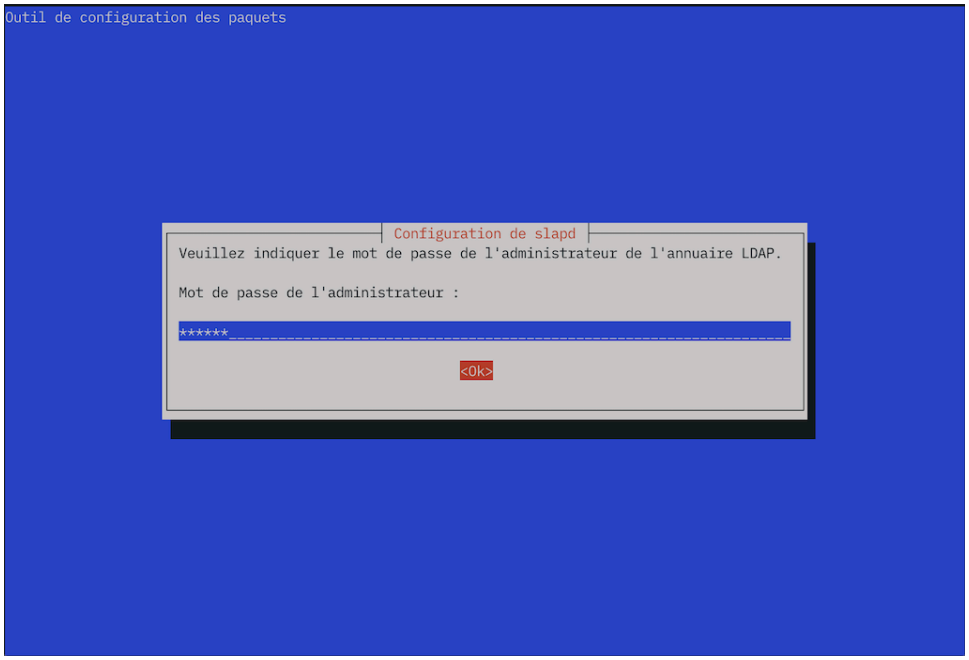
Copie d'écran dpkg-reconfigure slapd - vue complète



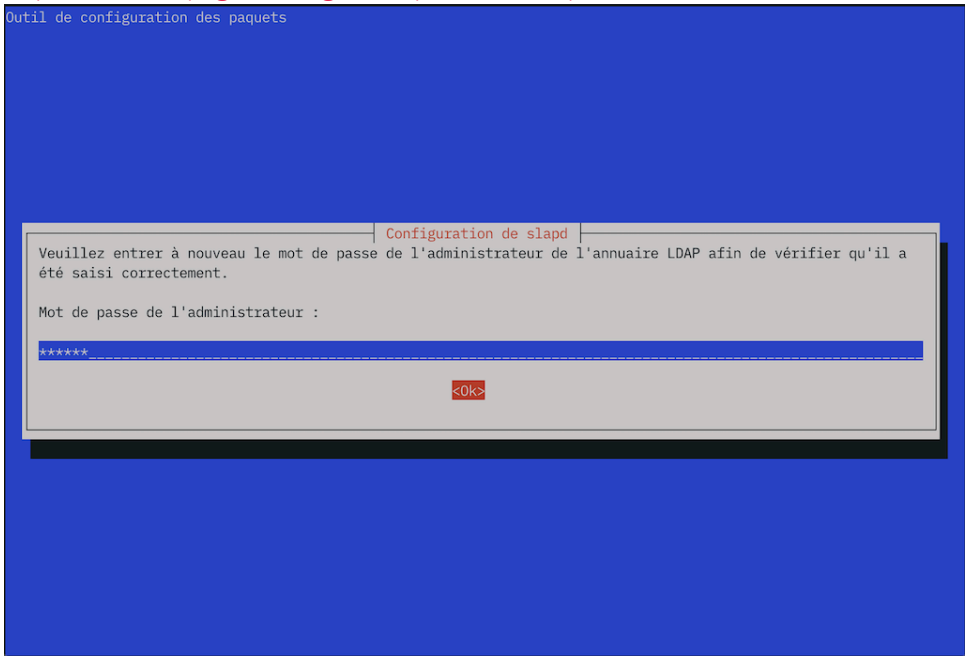
Copie d'écran dpkg-reconfigure slapd - vue complète



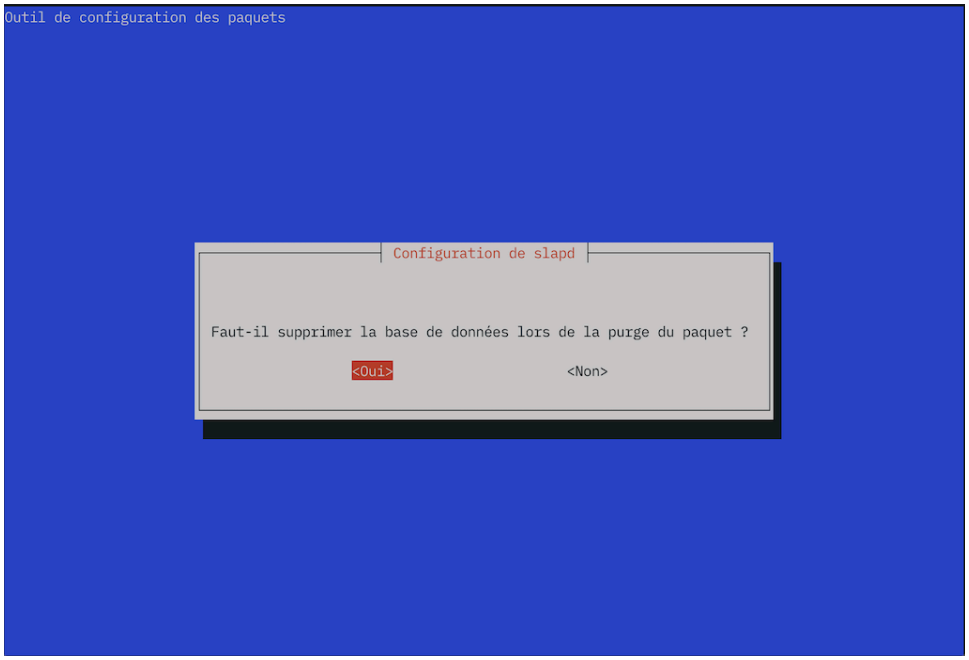
Copie d'écran dpkg-reconfigure slapd - vue complète



Copie d'écran dpkg-reconfigure slapd - vue complète



Copie d'écran dpkg-reconfigure slapd - vue complète



Copie d'écran dpkg-reconfigure slapd - vue complète

Q97. Comment valider la nouvelle configuration du paquet `slapd` ?

Reprendre la question sur le nom distinctif de l'annuaire.

```
sudo ldapsearch -LLL -Y EXTERNAL -H ldapi:/// -b "cn=config" \
  olcSuffix | grep ^olcSuffix
```

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
olcSuffix: dc=lab,dc=stri
```

2.4. Composition d'un nouvel annuaire LDAP

Une fois que les fichiers de configuration et de base de données du nouvel annuaire sont en place, on peut passer à l'ajout de nouvelles entrées dans cet annuaire. Comme le fil conducteur de cette série de travaux pratiques est la gestion d'une base de comptes utilisateurs, on doit ajouter les objets suivants.

- Deux unités organisationnelles : `people` et `groups`.
- Quatre compte utilisateurs : papa et maman Skywalker ainsi que leurs deux enfants

Toutes les manipulations sur les objets de l'annuaire utilisent un format de fichier texte particulier baptisé LDIF pour *LDAP Data Interchange Format*. C'est un format de représentation des données contenues dans un annuaire particulièrement utile pour les opérations de sauvegarde et de restauration en volume.

Du point de vue formatage, chaque enregistrement doit être séparé du suivant par une ligne vide et chaque attribut d'un enregistrement apparaît sur une ligne sous la forme «nomAttribut: valeur».

Q98. Comment visualiser la liste des entrées contenues dans l'annuaire LDAP ?

Utiliser les pages de manuels de la commande ldapsearch et rechercher les informations sur les méthodes d'authentification, la désignation de la base dans laquelle on effectue la recherche et le nom distinctif utilisé pour se connecter à l'annuaire.

La commande ldapsearch propose plusieurs modes d'authentification qui influent sur la liste des attributs affichés pour une même entrée. Dans notre exemple, ce sont les mots de passes qui peuvent ne pas apparaître ou apparaître sous différentes formes.

- L'option `-x` évite le recours à la méthode SASL pour l'authentification.

```
sudo ldapsearch -LLL -x -H ldap:/// -b "dc=lab,dc=stri" \
  -D cn=admin,dc=lab,dc=stri -W
```

```
Enter LDAP Password:
dn: dc=lab,dc=stri
objectClass: top
objectClass: dcObject
objectClass: organization
o: lab.stri
dc: lab
```

- L'option `-Y EXTERNAL` utilise la méthode SASL du même nom.

```
sudo ldapsearch -LLL -Y EXTERNAL -H ldapi:/// -b "dc=lab,dc=stri" \
  -D cn=admin,dc=lab,dc=stri -W
```

```
Enter LDAP Password:
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
dn: dc=lab,dc=stri
objectClass: top
objectClass: dcObject
objectClass: organization
o: lab.stri
dc: lab
```

- L'option `-LLL` désactive l'affichage des commentaires et de la version LDIF utilisée dans la réponse.
- L'option `-b` désigne le point de départ de la recherche.
- L'option `-D` désigne le nom distinctif de connexion à l'annuaire.
- L'option `-W` provoque l'affichage de l'invite de demande du mot passe correspondant au nom distinctif utilisé.

Q99. Comment activer la journalisation des manipulations sur les entrées de l'annuaire LDAP ?

Rechercher l'entrée relative au niveau de journalisation dans le DIT et modifier sa valeur de façon à obtenir un état dans les journaux système à chaque opération sur l'annuaire.

La modification de l'entrée du DIT doit se faire à l'aide d'un fichier LDIF approprié.

L'entrée à rechercher dans le DIT est baptisée `olcLogLevel`.

```
sudo ldapsearch -LLL -Y EXTERNAL -H ldapi:/// -b "cn=config" \
  olcLogLevel | grep ^olcLogLevel
```

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
olcLogLevel: none
```

on se propose de remplacer la valeur `none` par `stats` de façon à journaliser les connexions, les opérations et les résultats. Voici une copie du fichier LDIF permettant de réaliser cette modification.

On commence par créer un dossier dédié aux fichiers LDIF.

```
mkdir -p $HOME/ldif && cd $HOME/ldif
```

Ensuite on peut créer le fichier LDIF de modification de la journalisation du service LDAP.

```
cat > setolcLogLevel2stats.ldif << EOF
# Set olcLogLevel to "stats"
dn: cn=config
changetype: modify
replace: olcLogLevel
olcLogLevel: stats
EOF
```

On applique ce changement de valeur avec la commande ldapmodify puis on vérifie que l'attribut a bien reçu le paramètre.

```
sudo ldapmodify -Y EXTERNAL -H ldapi:/// -f setolcLogLevel2stats.ldif
```

```
CSASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
modifying entry "cn=config"
```

```
sudo ldapsearch -LLL -Y EXTERNAL -H ldapi:/// -b "cn=config" \
  olcLogLevel | grep ^olcLogLevel
```

```
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
olcLogLevel: stats
```

Enfin, on relève les traces de la dernière opération dans les journaux système.

```
journalctl -o cat -n 20 -u slapd --grep="conn"
conn=1009 fd=12 closed
conn=1009 op=2 UNBIND
conn=1009 op=1 SEARCH RESULT tag=101 err=0 qtime=0.000017 etime=0.000193 nentries=10 text=
conn=1009 op=1 SRCH attr=olcLogLevel
conn=1009 op=1 SRCH base="cn=config" scope=2 deref=0 filter="(objectClass=*)"
conn=1009 op=0 RESULT tag=97 err=0 qtime=0.000018 etime=0.000107 text=
conn=1009 op=0 BIND dn="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth" mech=EXTERNAL bind_ssf=0 ssf=71
conn=1009 op=0 BIND authcid="gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth" authzid="gidNumber=0+uidNumber=0,cn=p
conn=1009 op=0 BIND dn="" method=163
conn=1009 fd=12 ACCEPT from PATH=/var/run/slapd/ldapi (PATH=/var/run/slapd/ldapi)
conn=1008 fd=12 closed
conn=1008 op=2 UNBIND
conn=1008 op=1 RESULT tag=103 err=0 qtime=0.000021 etime=0.000558 text=
```

Note

Dans le contexte des travaux pratiques, le nombre d'entrées de l'annuaire reste très limité et la journalisation n'a pas d'impact mesurable sur les performances du système. Dans un contexte d'exploitation réelle avec un annuaire comprenant au moins une dizaine de milliers d'entrées, la situation est très différente et il faut limiter au maximum le recours à la journalisation des transactions sur l'annuaire.

Pour ramener la valeur de l'attribut `olcLogLevel` à `none`, il suffit de créer un fichier LDIF avec la directive correspondante.

```
cat > setolcLogLevel2none.ldif << EOF
# Set olcLogLevel to "none"
dn: cn=config
changetype: modify
replace: olcLogLevel
olcLogLevel: none
EOF
```

Q100. Quelle est la syntaxe du fichier LDIF qui permet d'ajouter les deux unités organisationnelles (***organisational unit***) ?

Rechercher un tutoriel LDIF en ligne donnant un exemple de fichier LDIF avec une ou plusieurs entrées `ou:`.

Voici un exemple de fichier LDIF contenant les déclarations des deux unités organisationnelles à ajouter.

```
cat > ou.ldif << EOF
dn: ou=people,dc=lab,dc=stri
objectClass: organizationalUnit
ou: people

dn: ou=groups,dc=lab,dc=stri
objectClass: organizationalUnit
ou: groups
EOF
```

Q101. Quelle est la commande à utiliser pour ajouter une ou plusieurs entrées dans l'annuaire ?

Rechercher dans la liste des programmes fournis avec le paquet des outils LDAP.

C'est la commande ldapadd qui est utile dans notre contexte. On l'utilise en mode d'authentification simple avec le fichier LDIF ci-dessus pour compléter l'annuaire.

```
sudo ldapadd -cxWD cn=admin,dc=lab,dc=stri -f ou.ldif
```

```
Enter LDAP Password:
adding new entry "ou=people,dc=lab,dc=stri"

adding new entry "ou=groups,dc=lab,dc=stri"
```

On vérifie ensuite que les deux nouvelles entrées sont bien présentes dans l'annuaire.

```
sudo ldapsearch -LLL -x -H ldap:/// -b "dc=lab,dc=stri" \
  -D cn=admin,dc=lab,dc=stri -W
```

```
Enter LDAP Password:
dn: dc=lab,dc=stri
objectClass: top
objectClass: dcObject
objectClass: organization
o: lab.stri
dc: lab

dn: ou=people,dc=lab,dc=stri
objectClass: organizationalUnit
ou: people

dn: ou=groups,dc=lab,dc=stri
objectClass: organizationalUnit
ou: groups
```

Q102. Quelle est la commande à utiliser pour saisir manuellement un mot de passe et obtenir la chaîne chiffrée correspondante ?

Rechercher dans la liste des programmes fournis avec les paquets de la distribution puis consulter les pages de manuels correspondantes.

En effectuant une recherche par mot clé dans les pages de manuels du système, on peut identifier l'outil recherché.

```
man -k passwd | grep -i ldap
```

```
ldappasswd (1)       - change the password of an LDAP entry
slappasswd (8)       - OpenLDAP password utility
```

On utilise la commande slappasswd pour générer une chaîne chiffrée que l'on insère dans le fichier LDIF des comptes utilisateurs.

Prenons l'exemple du mot de passe `V3ry53cr3t`, on obtient le résultat suivant :

```
sudo slappasswd
```

```
New password:
Re-enter new password:
{SSHA}rpB4tgcVlh51sPCtpBi+acrS6Ifc1lu0
```

Dans le contexte de ces travaux pratiques, on attribue le même mot de passe aux quatre comptes utilisateurs.

Il existe une technique simple pour la génération de mots de passe utilisateurs aléatoires. Une fois le mot de passe généré, il peut être transmis à l'utilisateur final par un «canal de confiance» et implanté dans les attributs de l'annuaire relatifs au compte utilisateur.

1. On génère un mot de passe aléatoire que l'on stocke dans un fichier.

   ```
   openssl rand -base64 16 | tr -d '=' > user.passwd
   ```

   On obtient par exemple :

   ```
   cat user.passwd
   vyJtXX6r73KPzyDYymWjsA
   ```

2. Utilise ce mot de passe pour générer la chaîne à introduire dans le fichier LDIF de création d'utilisateur dans l'annuaire.

   ```
   sudo slappasswd -v -h "{SSHA}" -s $(cat user.passwd)
   ```

   ```
   {SSHA}hFGouu+ytfnH0qPy7y9G0L0Rb6R6slZ4
   ```

Q103. Quelle est la syntaxe du fichier LDIF qui permet d'ajouter les quatre utilisateurs avec leurs attributs système : identifiants `uid/gid`, authentifiants `login/passwd`, etc ?

Rechercher un tutoriel LDIF en ligne donnant un exemple de fichier LDIF avec un exemple de description des attributs d'un compte utilisateur.

Voici un exemple de fichier LDIF contenant les déclarations des quatre comptes utilisateurs à ajouter.

Avertissement

Pensez à adapter les entrées `userPassword` à votre contexte !

```
cat > users.ldif << EOF
# Padmé Amidala
dn: uid=padme,ou=people,dc=lab,dc=stri
objectClass: inetOrgPerson
objectClass: shadowAccount
objectClass: posixAccount
cn: Padme
sn: Padmé Amidala Skywalker
uid: padme
uidNumber: 10000
gidNumber: 10000
loginShell: /bin/bash
homeDirectory: /ahome/padme
userPassword: {SSHA}hFGouu+ytfnH0qPy7y9G0L0Rb6R6slZ4
gecos: Padme Amidala Skywalker

# Anakin Skywalker
dn: uid=anakin,ou=people,dc=lab,dc=stri
objectClass: inetOrgPerson
objectClass: shadowAccount
objectClass: posixAccount
cn: Anakin
sn: Anakin Skywalker
uid: anakin
uidNumber: 10001
gidNumber: 10001
loginShell: /bin/bash
homeDirectory: /ahome/anakin
userPassword: {SSHA}hFGouu+ytfnH0qPy7y9G0L0Rb6R6slZ4
gecos: Anakin Skywalker

# Leia Organa Skywalker
dn: uid=leia,ou=people,dc=lab,dc=stri
objectClass: inetOrgPerson
objectClass: shadowAccount
objectClass: posixAccount
cn: Leia
sn: Leia Organa
uid: leia
uidNumber: 10002
gidNumber: 10002
loginShell: /bin/bash
homeDirectory: /ahome/leia
userPassword: {SSHA}hFGouu+ytfnH0qPy7y9G0L0Rb6R6slZ4
gecos: Leia Organa Skywalker

# Luke Skywalker
dn: uid=luke,ou=people,dc=lab,dc=stri
objectClass: inetOrgPerson
objectClass: shadowAccount
objectClass: posixAccount
cn: Luke
sn: Luke Skywalker
uid: luke
uidNumber: 10003
gidNumber: 10003
loginShell: /bin/bash
homeDirectory: /ahome/luke
userPassword: {SSHA}hFGouu+ytfnH0qPy7y9G0L0Rb6R6slZ4
gecos: Luke Skywalker
EOF
```

Comme dans le cas précédent, on utilise la commande ldapadd en mode d'authentification simple pour insérer les utilisateurs dans l'annuaire.

```
sudo ldapadd -cxWD cn=admin,dc=lab,dc=stri -f users.ldif
```

```
Enter LDAP Password:
adding new entry "uid=padme,ou=people,dc=lab,dc=stri"

adding new entry "uid=anakin,ou=people,dc=lab,dc=stri"

adding new entry "uid=leia,ou=people,dc=lab,dc=stri"

adding new entry "uid=luke,ou=people,dc=lab,dc=stri"
```

On peut lister à nouveau les entrées contenues dans l'annuaire pour vérifier la présence des utilisateurs.

```
sudo ldapsearch -LLL -x -H ldap:/// -b "dc=lab,dc=stri" \
  -D cn=admin,dc=lab,dc=stri -W
```

3. Configuration de l'accès client au serveur LDAP

Dans cette section, on suppose qu'un annuaire LDAP existe et qu'il contient des utilisateurs. On se propose de configurer un poste client pour qu'il obtienne de façon transparente les informations sur les comptes utilisateurs desservis par l'annuaire.

3.1. Interrogation à distance de l'annuaire LDAP

On reprend ici les requêtes de consultation des entrées de l'annuaire vues dans la Section 2.4, « Composition d'un nouvel annuaire LDAP ». Cette fois-ci les requêtes sont émises depuis un hôte réseau différent du serveur LDAP.

Q104. Quel est le paquet qui fournit, entre autres, la commande de consultation des entrées de l'annuaire ?

Interroger la base de données des paquets pour obtenir les informations demandées.

```
sudo apt -y install ldap-utils
```

Le paquet `ldap-utils` apparaît à la question sur la liste des paquets relatifs au service LDAP. Si on recherche les commandes présentes dans la liste des fichiers de ce paquet, on obtient les informations suivantes.

```
dpkg -L ldap-utils | grep "bin/"
```

```
/usr/bin/ldapcompare
/usr/bin/ldapdelete
/usr/bin/ldapexop
/usr/bin/ldapmodify
/usr/bin/ldapmodrdn
/usr/bin/ldappasswd
/usr/bin/ldapsearch
/usr/bin/ldapurl
/usr/bin/ldapwhoami
/usr/bin/ldapadd
```

Une fois ce paquet installé, il est possible d'utiliser toutes les commandes disponibles pour manipuler les enregistrements de l'annuaire.

Q105. Quelle est la syntaxe d'interrogation de l'annuaire qui permet d'obtenir tous les attributs de l'enregistrement correspondant à un utilisateur particulier ?

On utilise la commande ldapsearch en spécifiant un attribut `uid` particulier.

```
sudo ldapsearch -LLL -H ldap://[2001:678:3fc:64:baad:caff:fefe:7] \
  -b dc=lab,dc=stri -D cn=admin,dc=lab,dc=stri -W uid=padme
```

```
Enter LDAP Password:
dn: uid=padme,ou=people,dc=lab,dc=stri
objectClass: inetOrgPerson
objectClass: shadowAccount
objectClass: posixAccount
cn: Padme
sn:: UGFkbcOpIEFtaWRhbGEgU2t5d2Fsa2Vy
uid: padme
uidNumber: 10000
gidNumber: 10000
loginShell: /bin/bash
homeDirectory: /ahome/padme
userPassword:: e1NTSEF9aEZHb3V1K3l0Zm5IMHFQeTd5OUcwTDBSYjZSNnNsWjQ=
gecos: Padme Amidala Skywalker
```

Q106. Quelle est la syntaxe de la commande permettant de changer le mot de passe de l'utilisateur dont on a affiché les attributs à la question précédente ?

On utilise la commande ldappasswd fournie par le paquet `ldap-utils` comme dans le cas de la commande de recherche. Après consultation des pages de manuels, on obtient la syntaxe suivante.

```
sudo ldappasswd -x -H ldap://[2001:678:3fc:64:baad:caff:fefe:7] \
  -D cn=admin,dc=lab,dc=stri -W -S uid=padme,ou=people,dc=lab,dc=stri
```

```
New password:
Re-enter new password:
Enter LDAP Password:
```

En posant exactement la même requête que dans la question précédente, on peut vérifier que le mot de passe utilisateur a bien été modifié.

```
sudo ldapsearch -LLL -H ldap://[2001:678:3fc:64:baad:caff:fefe:7] \
  -b dc=lab,dc=stri -D cn=admin,dc=lab,dc=stri -W uid=padme
```

```
Enter LDAP Password:
dn: uid=padme,ou=people,dc=lab,dc=stri
objectClass: inetOrgPerson
objectClass: shadowAccount
objectClass: posixAccount
cn: Padme
sn:: UGFkbcOpIEFtaWRhbGEgU2t5d2Fsa2Vy
uid: padme
uidNumber: 10000
gidNumber: 10000
loginShell: /bin/bash
homeDirectory: /ahome/padme
gecos: Padme Amidala Skywalker
userPassword:: e1NTSEF9bngwTTlpUi9QYitpaVJTbzNpN0tkejVkSTRJMVpZclM=
```

3.2. Configuration *Name Service Switch*

Les manipulations présentées ici ont pour but de rendre transparent l'accès aux attributs des comptes utilisateurs. Le mécanisme *Name Service Switch* assure un aiguillage de l'accès à ces attributs entre les fichiers locaux et les différents services réseau disponibles. Ici, l'annuaire LDAP constitue un dépôt de référence pour le stockage des attributs des comptes utilisateurs.

Q107. Quel est le nom du paquet relatif au mécanisme *Name Service Switch* permettant d'accéder aux ressources de l'annuaire LDAP ?

Rechercher dans les bases du gestionnaire de paquets un paquet dont le nom débute par la chaîne `libnss`.

La liste ci-dessous permet d'identifier le paquet `libnss-ldapd`.

```
apt search --names-only ^libnss-
```

```
apt search --names-only ^libnss-ldap
```

```
En train de trier... Fait
Recherche en texte intégral... Fait
libnss-ldapd/testing 0.9.12-4 amd64
  NSS module for using LDAP as a naming service
```

Q108. Quels sont les paquets supplémentaires qui sont ajoutés lors de l'installation des bibliothèques LDAP pour le mécanisme *Name Service Switch* ?

Utiliser les informations fournies par le gestionnaire de paquets pour chaque ajout.

Le lancement de l'installation du paquet `libnss-ldapd` donne la liste suivante.

```
sudo apt install libnss-ldapd
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  libpam-ldapd nscd nslcd nslcd-utils
Paquets suggérés :
  kstart
Les NOUVEAUX paquets suivants seront installés :
  libnss-ldapd libpam-ldapd nscd nslcd nslcd-utils
0 mis à jour, 5 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 390 ko dans les archives.
Après cette opération, 971 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n]
```

Plusieurs paquets supplémentaires apparaissent :

- `libpam-ldapd` fournit les fonctions PAM nécessaires à l'authentification, aux autorisations et à la gestion de session via un annuaire LDAP.
- `nscd` (*Name Service Cache Daemon*) est un démon qui gère la recherche des mots de passe, des groupes et hôtes des programmes en cours d'exécution, et met en cache le résultat pour une prochaine recherche.
- `nslcd` fournit un autre démon pour la collecte des informations sur les comptes utilisateurs depuis un serveur LDAP.
- `nslcd-utils` fournit des outils pour l'interrogation et la mise à jour des entrées d'annuaire LDAP.

Avertissement

Pour les besoins des travaux pratiques ou de la mise au point de l'authentification via LDAP, il est utile de relancer les services de cache à chaque modification des conditions d'accès à l'annuaire.

```
sudo systemctl restart nslcd
```

```
sudo systemctl restart nscd
```

Q109. Quel est le rôle de l'interface entre les fonctions PAM (***Pluggable Authentication Modules***) et l'annuaire LDAP ?

Par définition, PAM est un mécanisme qui permet d'intégrer différents modes d'authentification en les rendant transparents vis à vis de l'utilisateur et des logiciels qui accèdent aux ressources du système. Dans le contexte de ces travaux pratiques, il s'agit de permettre à l'utilisateur de se connecter, d'accéder au système de fichiers, de changer son mot de passe, etc sans avoir à lancer des commandes spécifiques.

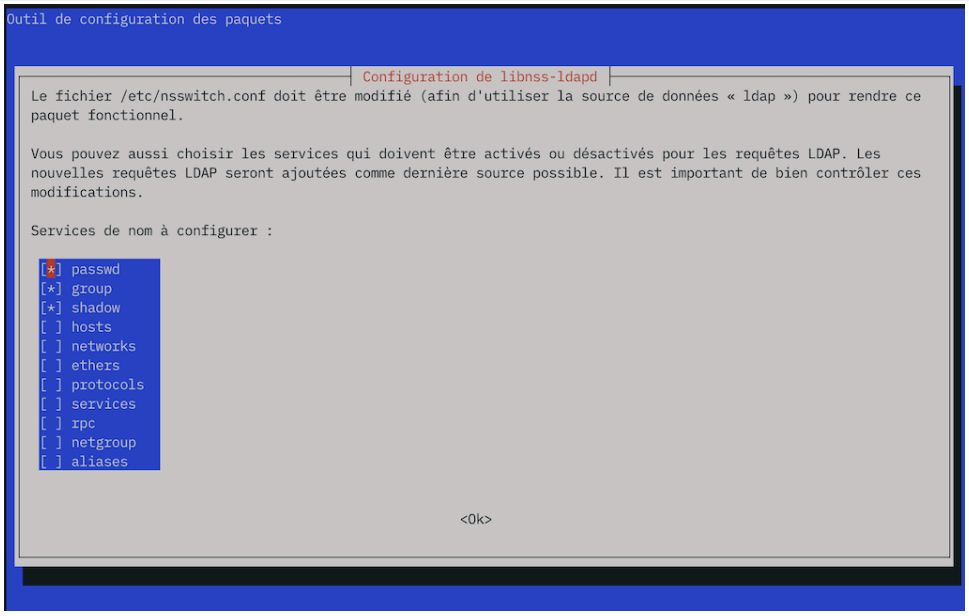
Q110. Quelles sont les principales étapes de la configuration des paquets de bibliothèques NSS et PAM ?

Lors de l'installation des principaux paquets de bibliothèques LDAP, on passe par une série de menus `debconf` qu'il faut renseigner correctement pour accéder au serveur LDAP de façon transparente.

Avertissement

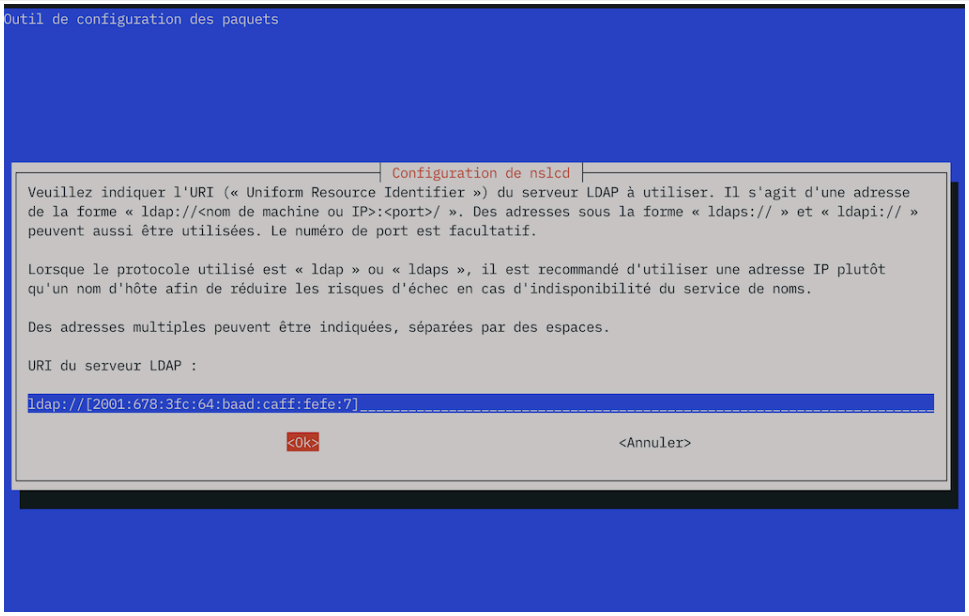
En cas d'erreur de saisie dans la série de menus ci-dessous, il faut reprendre la configuration de chacun des deux paquets individuellement. Classiquement, on passe par la commande dpkg-reconfigure.

```
sudo dpkg-reconfigure libnss-ldapd
```

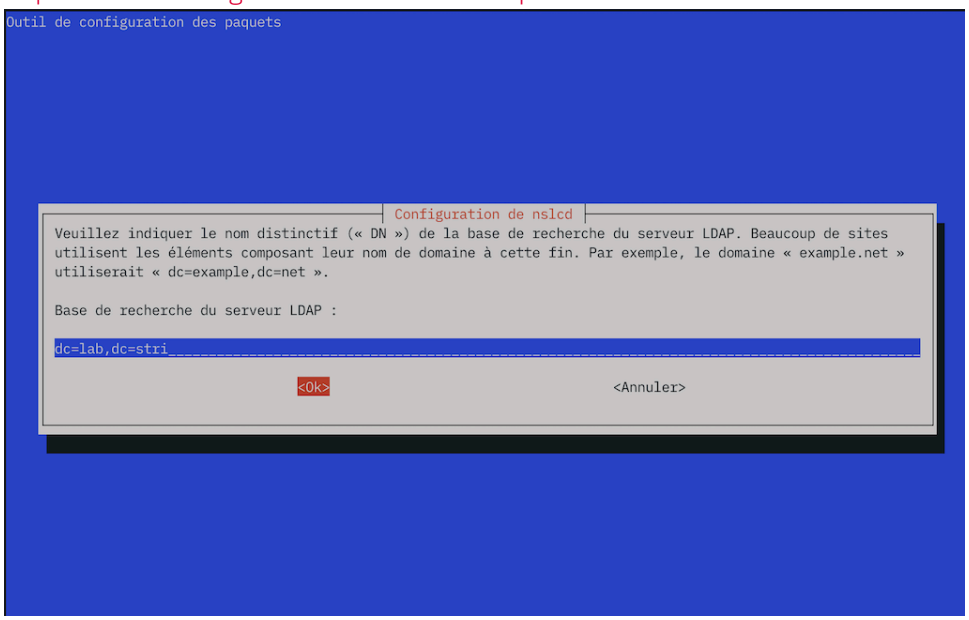


Copie d'écran configuration libnss-ldapd - vue complète

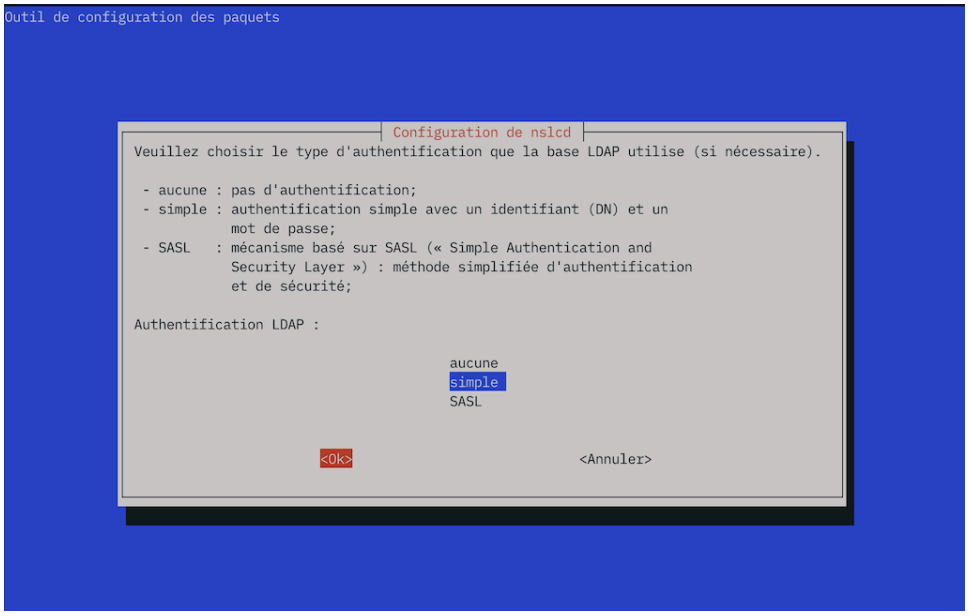
```
sudo dpkg-reconfigure nslcd
```



Copie d'écran configuration nslcd - vue complète

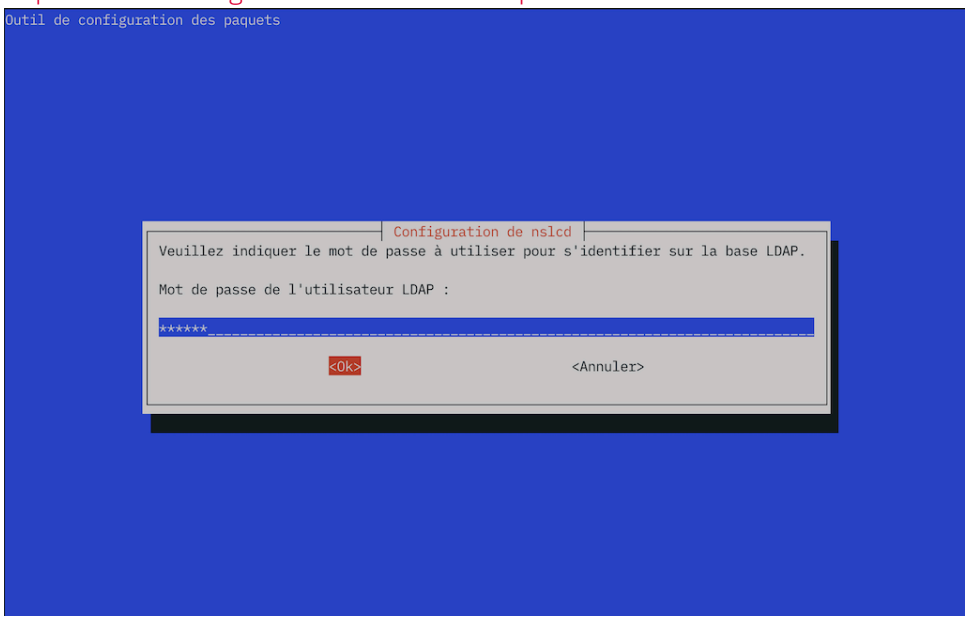


Copie d'écran configuration nslcd - vue complète

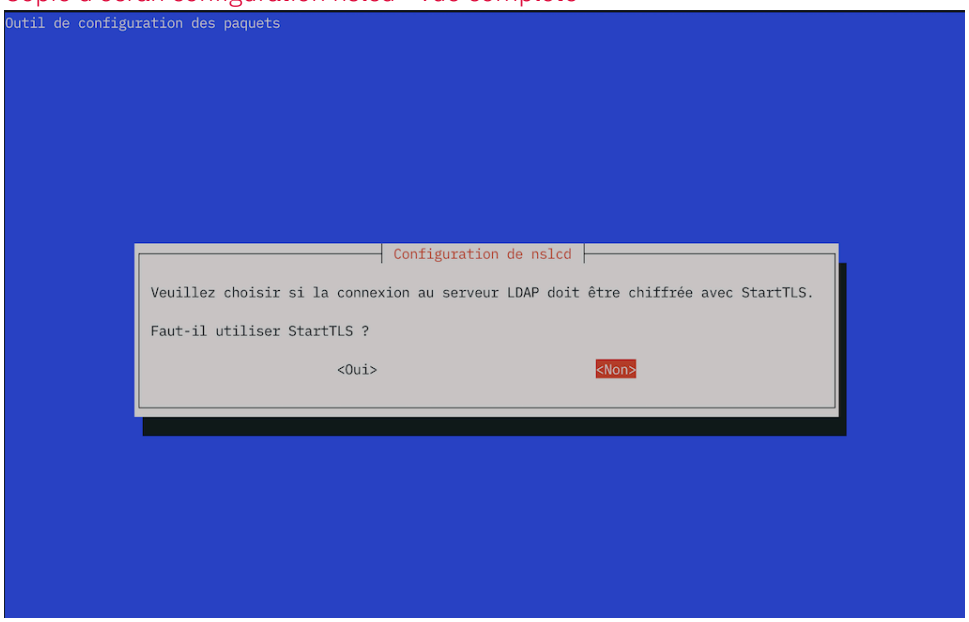


Copie d'écran configuration nslcd - vue complète

Copie d'écran configuration nslcd - vue complète



Copie d'écran configuration nslcd - vue complète



Copie d'écran configuration nslcd - vue complète

Q111. Quelles sont les modifications apportées au fichier de configuration `/etc/nsswitch.conf` pour activer l'accès aux ressources de l'annuaire LDAP ?

Lors de l'installation des paquets à l'étape précédente, le fichier `/etc/nsswitch.conf` a été modifié.

```
grep ldap /etc/nsswitch.conf
passwd:         files systemd ldap
group:          files systemd ldap
shadow:         files systemd ldap
```

Q112. Comment illustrer simplement le fonctionnement du mécanisme ***name service switch*** intégrant l'utilisation de l'annuaire LDAP ?

Rechercher la commande de récupération des entrées depuis les bases de données d'administration dans les outils fournis avec les bibliothèques standard (paquet `libc-bin`).

```
dpkg -L libc-bin | grep "bin/"
```

La commande getent fournie avec le paquet `libc-bin` donne la liste des entrées accessibles pour chaque catégorie du fichier de configuration. Voici un exemple pour la catégorie `passwd` qui fait apparaître les entrées de l'annuaire LDAP à la suite des comptes utilisateurs système issus des fichiers locaux.

```
getent passwd
```

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
systemd-timesync:x:997:997:systemd Time Synchronization:/:/usr/sbin/nologin
messagebus:x:100:107::/nonexistent:/usr/sbin/nologin
sshd:x:101:65534::/run/sshd:/usr/sbin/nologin
etu:x:1000:1000:Etudiant.e,,,:/home/etu:/bin/bash
systemd-resolve:x:996:996:systemd Resolver:/:/usr/sbin/nologin
rdnssd:x:102:65534::/var/run/rdnssd:/usr/sbin/nologin
nslcd:x:103:109:nslcd name service LDAP connection daemon,,,:/run/nslcd:/usr/sbin/nologin
padme:x:10000:10000:Padme Amidala Skywalker:/ahome/padme:/bin/bash
anakin:x:10001:10001:Anakin Skywalker:/ahome/anakin:/bin/bash
leia:x:10002:10002:Leia Organa Skywalker:/ahome/leia:/bin/bash
luke:x:10003:10003:Luke Skywalker:/ahome/luke:/bin/bash
```

Q113. Comment valider l'authentification d'un utilisateur déclaré dans l'annuaire LDAP ?

Choisir un service qui nécessite une authentification sur le système et qui utilise une entrée de l'annuaire LDAP.

Les exemples de services nécessitant une authentification ne manquent pas. La commande su qui permet de changer d'identité est le plus immédiat.

```
su - padme
```

```
Mot de passe :
su: avertissement : impossible de changer le répertoire vers /ahome/padme: Aucun fichier ou dossier de ce type
padme@ldap-client:/home/etu$
```

Dans les journaux du système, on retrouve les mêmes éléments.

```
journalctl -o cat -n 20 --grep="pam_unix" | grep padme
```

```
pam_unix(su-l:session): session closed for user padme
pam_unix(su-l:session): session opened for user padme(uid=10000) by etu(uid=1000)
pam_unix(su-l:auth): authentication failure; logname=etu uid=1000 euid=0 tty=/dev/pts/0 ruser=etu rhost=  user=padme
pam_unix(su-l:session): session closed for user padme
pam_unix(su-l:session): session opened for user padme(uid=10000) by etu(uid=1000)
pam_unix(su-l:auth): authentication failure; logname=etu uid=1000 euid=0 tty=/dev/pts/0 ruser=etu rhost=  user=padme
```

Voici un autre exemple d'accès avec ssh.

```
ssh padme@localhost
```

```
The authenticity of host 'localhost (::1)' can't be established.
ED25519 key fingerprint is SHA256:yFLaZk+OfY7z7bHyHPXgjowRS4KMHjfoMQxracRdG9M.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
padme@localhost's password:
Linux ldap-client 6.4.0-3-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.4.11-1 (2023-08-17) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Could not chdir to home directory /ahome/padme: No such file or directory
padme@ldap-client:/$
déconnexion
Connection to localhost closed.
```

```
journalctl -o cat -n 100 -u ssh | grep padme
```

Il ne manque que l'accès au système de fichiers pour que la configuration soit vraiment complète.

4. accès à l'annuaire LDAP depuis un service web

Du point de vue métier, les manipulations à base de fichiers LDIF sont réservées aux traitements en volume réalisés par les administrateurs système. Les développeurs disposent de bibliothèques fournies avec les langages de programmation. Dans la plupart des cas, les développements ont pour but de fournir une interface web.

Le projet *LDAP Tool Box project* propose un outil baptisé ***white pages*** qui permet de constituer un trombinoscope des utilisateurs enregistrés dans un annuaire LDAP.

l'objectif de cette section est d'installer le service web *White Pages* et de compléter les attributs des utilisateurs de l'annuaire avec une photo.

Q114. Quel est le paquet à installer pour mettre en place le service web ***White Pages*** ?

Rechercher sur le site *LDAP Tool Box project*, le lien de téléchargement direct du paquet Debian pour le service ***White Pages***.

À partir du lien `Download` en bas de la page principale, on trouve un lien direct vers le paquet.

Après le téléchargement, l'installation nécessite quelques ajustements compte tenu des dépendances des paquets entre les différentes versions du langage PHP et du ***framework Smarty***.

```
wget https://ltb-project.org/archives/white-pages_0.4-2_all.deb
```

```
sudo dpkg -i white-pages_0.4-2_all.deb
```

```
sudo apt -y -f install
```

```
sudo apt install smarty3
```

Q115. Comment activer l'accès au service web ?

Consulter les fichiers de documentation et de configuration fournis avec le paquet ***apache2***. Repérer les instructions d'activation et de désactivation d'un site. Retrouver les éléments spécifiques à la configuration du service ***White Pages***.

Cette question comprend plusieurs étapes.

1. Le paquet `apache2` comprend une liste d'outils dédiés aux manipulations sur les sites et leur configuration.

```
dpkg -L apache2 | grep "bin.*a2"
/usr/sbin/a2enmod
/usr/sbin/a2query
/usr/sbin/a2disconf
/usr/sbin/a2dismod
/usr/sbin/a2dissite
/usr/sbin/a2enconf
/usr/sbin/a2ensite
```

2. On utilise a2dissite pour désactiver le site par défaut et a2ensite pour activer les pages blanches.

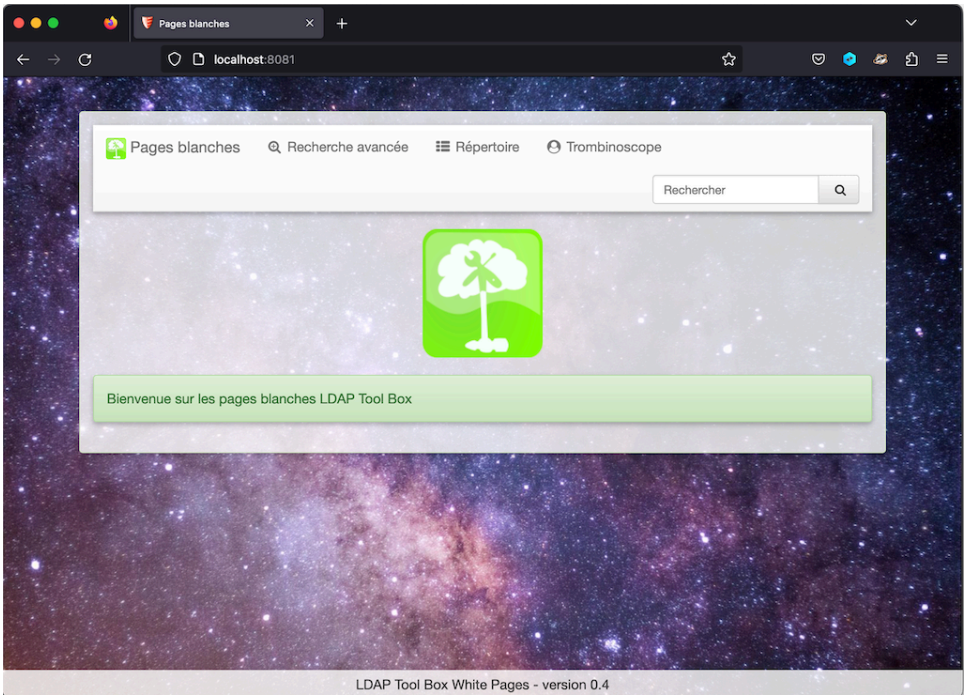
```
sudo a2dissite 000-default
```

```
sudo a2ensite white-pages
```

```
sudo apachectl configtest
```

```
sudo systemctl reload apache2
```

La consultation de la page web donne le résultat suivant.



Copie d'écran service pages blanches - vue complète

3. Les paramètres du nouveau site sont donnés dans le fichier `/etc/apache2/sites-available/white-pages.conf`.

Q116. Comment paramétrer l'accès à l'annuaire LDAP à partir du service web ?

Identifier les fichiers de configuration fournis avec le paquet `white-pages`.

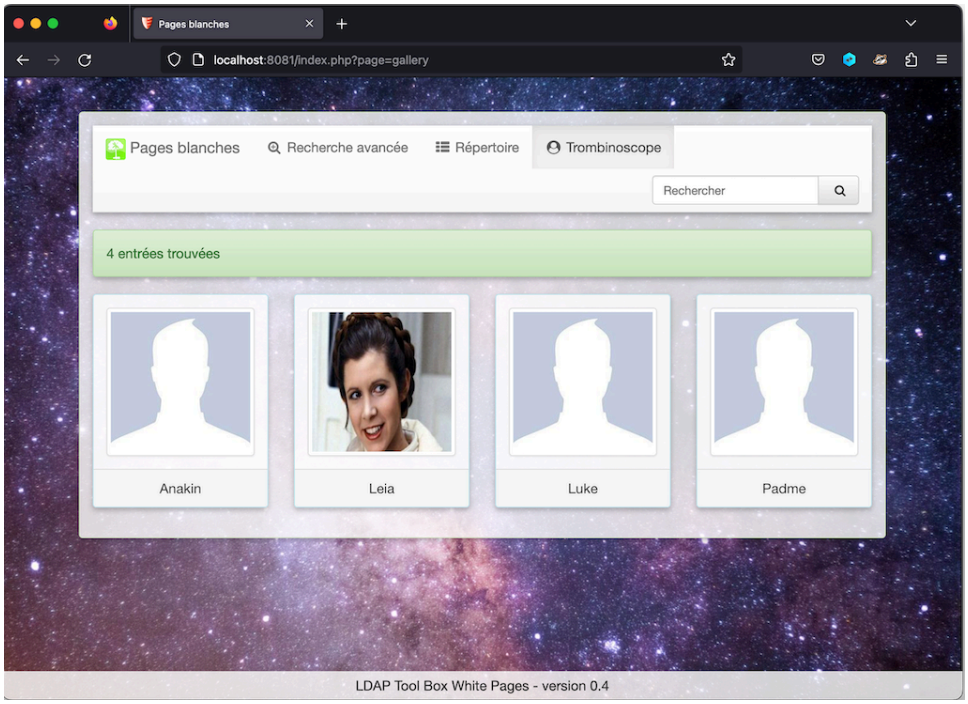
C'est le fichier `/usr/share/white-pages/conf/config.inc.php` qui contient les éléments d'accès à l'annuaire LDAP. Voici un extrait de ce fichier avec les lignes utiles complétées avec le contexte de ce support de travaux pratiques.

```
# grep ^\$ldap /usr/share/white-pages/conf/config.inc.php
$ldap_url = "ldap://localhost";
$ldap_starttls = false;
$ldap_binddn = "cn=admin,dc=lab,dc=stri";
$ldap_bindpw = "xxxxxx";
$ldap_base = "dc=lab,dc=stri";
$ldap_user_base = "ou=people,".$ldap_base;
$ldap_user_filter = "(objectclass=inetorgperson)";
$ldap_size_limit = 100;
```

Une fois le fichier modifié, il faut recharger la configuration du service web.

```
sudo systemctl reload apache2
```

La consultation de la rubrique pages blanches donne le résultat ci-dessous. L'intérêt de cette manipulation est de montrer que l'on peut compléter les attributs d'un utilisateur de l'annuaire avec une photo. Cette opération est l'objet des questions suivantes.



Copie d'écran trombinoscope - vue complète

Q117. Quel est l'attribut de la classe `inetorgperson` qui correspond à une photo d'identité ?

Rechercher les options de la commande ldapsearch qui permettent d'extraire la liste des attributs de la classe `inetorgperson`.

On obtient l'information en deux temps.

- On identifie le contexte de la classe recherchée en premier. Voici un exemple de requête côté serveur.

```
sudo ldapsearch -LLL -Y EXTERNAL -H ldapi:/// \
 -b "cn=config" | grep -i inetorgperson
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
dn: cn={3}inetorgperson,cn=schema,cn=config
cn: {3}inetorgperson
olcObjectClasses: {0}( 2.16.840.1.113730.3.2.2 NAME 'inetOrgPerson' DESC 'RFC2
```

- Une fois le contexte connu avec précision, on peut extraire la liste des attributs relatifs à la classe `inetorgperson`.

  Dans la liste ci-dessous, on repère l'attribut `jpegphoto` qui correspond à notre besoin.

```
sudo ldapsearch -LLL -Y EXTERNAL -H ldapi:/// \
 -b "cn={3}inetorgperson,cn=schema,cn=config"
SASL/EXTERNAL authentication started
SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
SASL SSF: 0
dn: cn={3}inetorgperson,cn=schema,cn=config
objectClass: olcSchemaConfig
cn: {3}inetorgperson
olcAttributeTypes: {0}( 2.16.840.1.113730.3.1.1 NAME 'carLicense' DESC 'RFC279
 8: vehicle license or registration plate' EQUALITY caseIgnoreMatch SUBSTR cas
 eIgnoreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
olcAttributeTypes: {1}( 2.16.840.1.113730.3.1.2 NAME 'departmentNumber' DESC '
 RFC2798: identifies a department within an organization' EQUALITY caseIgnoreM
 atch SUBSTR caseIgnoreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
olcAttributeTypes: {2}( 2.16.840.1.113730.3.1.241 NAME 'displayName' DESC 'RFC
 2798: preferred name to be used when displaying entries' EQUALITY caseIgnoreM
 atch SUBSTR caseIgnoreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SI
 NGLE-VALUE )
olcAttributeTypes: {3}( 2.16.840.1.113730.3.1.3 NAME 'employeeNumber' DESC 'RF
 C2798: numerically identifies an employee within an organization' EQUALITY ca
 seIgnoreMatch SUBSTR caseIgnoreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.12
 1.1.15 SINGLE-VALUE )
olcAttributeTypes: {4}( 2.16.840.1.113730.3.1.4 NAME 'employeeType' DESC 'RFC2
 798: type of employment for a person' EQUALITY caseIgnoreMatch SUBSTR caseIgn
 oreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
olcAttributeTypes: {5}( 0.9.2342.19200300.100.1.60 NAME 'jpegPhoto' DESC 'RFC2
 798: a JPEG image' SYNTAX 1.3.6.1.4.1.1466.115.121.1.28 )
olcAttributeTypes: {6}( 2.16.840.1.113730.3.1.39 NAME 'preferredLanguage' DESC
  'RFC2798: preferred written or spoken language for a person' EQUALITY caseIg
 noreMatch SUBSTR caseIgnoreSubstringsMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.
 15 SINGLE-VALUE )
olcAttributeTypes: {7}( 2.16.840.1.113730.3.1.40 NAME 'userSMIMECertificate' D
 ESC 'RFC2798: PKCS#7 SignedData used to support S/MIME' SYNTAX 1.3.6.1.4.1.14
 66.115.121.1.5 )
olcAttributeTypes: {8}( 2.16.840.1.113730.3.1.216 NAME 'userPKCS12' DESC 'RFC2
 798: personal identity information, a PKCS #12 PFX' SYNTAX 1.3.6.1.4.1.1466.1
 15.121.1.5 )
olcObjectClasses: {0}( 2.16.840.1.113730.3.2.2 NAME 'inetOrgPerson' DESC 'RFC2
 798: Internet Organizational Person' SUP organizationalPerson STRUCTURAL MAY
 ( audio $ businessCategory $ carLicense $ departmentNumber $ displayName $ em
 ployeeNumber $ employeeType $ givenName $ homePhone $ homePostalAddress $ ini
 tials $ jpegPhoto $ labeledURI $ mail $ manager $ mobile $ o $ pager $ photo
 $ roomNumber $ secretary $ uid $ userCertificate $ x500uniqueIdentifier $ pre
 ferredLanguage $ userSMIMECertificate $ userPKCS12 ) )
```

Q118. Quelle est la syntaxe du fichier LDIF qui permet de modifier l'attribut `jpegphoto` d'un utilisateur de l'annuaire ?

Rechercher un exemple de modification d'attribut avec la commande ldapmodify.

Rechercher aussi un fichier JPEG qui fasse office de photo d'identité.

Tout d'abord, on dépose le fichier jpeg à utiliser dans le dossier `/var/tmp` à titre d'exemple.

```
ls -l /var/tmp/leia.jpg
 -rw-r--r-- 1 etu etu 83837 19 août  03:15 /var/tmp/leia.jpg
```

La syntaxe du fichier LDIF est relativement simple une fois que l'on a bien identifié le contexte.

```
cat > leia-photo.ldif << EOF
dn: uid=leia,ou=people,dc=lab,dc=stri
changetype: modify
add: jpegphoto
jpegphoto:<file:///var/tmp/leia.jpg
EOF
```

Enfin, on applique la modification dans l'annuaire LDAP.

```
sudo ldapmodify -x -H ldap:/// -D "cn=admin,dc=lab,dc=stri" -W -f leia-photo.ldif
```

Le résultat est visible sur la copie d'écran de navigateur web ci-dessus.

5. Sécurisation des échanges avec TLS

Partant d'un service LDAP fonctionnel, nous allons maintenant sécuriser les échanges entre le serveur et ses clients en utilisant la sécurité de couche transport ou *Transport Layer Security* (TLS).

Dans ce but, nous devons installer et configurer une autorité de certification locale dans ce contexte de travaux pratiques.

En "situation réelle", on ferait appel à une autorité de certification tierce publique comme *Let's Encrypt*.

5.1. Génération des certificats avec easyrsa

Cette étape débute par l'installation du paquet `easy-rsa`, l'initialisation d'une nouvelle autorité (CA) et la génération d'un paire de clés.

Une fois le paquet `easy-rsa` installé, toutes les opérations de mise en place de l'autorité de certification se font à partir d'une session administrateur. C'est la raison de la présence de la commande sudo -i ci-dessous.

1. Installation du paquet.

```
sudo apt install easy-rsa
```

2. Création de l'arborescence de l'autorité de certification.

```
sudo -i
make-cadir ldap-pki
cd ldap-pki
```

```
root@ldap-server:~/ldap-pki# ls -lAh
total 20K
lrwxrwxrwx 1 root root   27  6 sept. 18:54 easyrsa -> /usr/share/easy-rsa/easyrsa
-rw-r--r-- 1 root root 5,1K  6 sept. 18:54 openssl-easyrsa.cnf
-rw-r--r-- 1 root root 8,9K  6 sept. 18:54 vars
lrwxrwxrwx 1 root root   30  6 sept. 18:54 x509-types -> /usr/share/easy-rsa/x509-types
```

3. Initialisation du gestionnaire de clés.

```
./easyrsa init-pki
```

4. Construction de l'autorité de certification.

```
./easyrsa build-ca nopass
```

5. Génération des certificats

```
./easyrsa build-server-full ldap.lab.stri nopass
```

6. documents de référence

OpenLDAP software 2.6 administrator's guide

La documentation officielle : *OpenLDAP Software 2.6 Administrator's Guide* constitue le point d'entrée essentiel pour la mise en œuvre du service LDAP.

Association LDAP, NFSv4 et autofs

https://www.inetdoc.net

Résumé

Ce support reprend les deux précédents sur NFSv4 et LDAP en associant les services. Le système de fichiers réseau NFSv4 sert au partage des répertoires utilisateur tandis que l'annuaire LDAP sert au partage des attributs des comptes utilisateur et de la configuration du service d'automontage. Une fois que les deux services associés sont en place, les comptes utilisateurs peuvent être utilisés de façon transparente depuis n'importe quel poste client faisant appel à ces services.

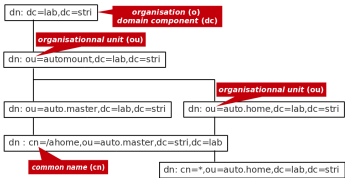


Table des matières

1. Mise en œuvre de l'annuaire LDAP

Cette partie reprend les étapes décrites dans le support *Introduction aux annuaires LDAP avec OpenLDAP*. Il s'agit d'installer les paquets correspondants au logiciel *OpenLDAP*, d'initialiser une base avec le bon contexte de nommage puis d'implanter les deux unités organisationnelles et les entrées des comptes utilisateurs.

Q119. Comment installer le service d'annuaire LDAP sur le poste serveur ?

Choisir les paquets à installer et valider le bon fonctionnement du service en contrôlant la liste des processus et des numéros de ports ouverts.

Reprendre les questions des parties Installation du serveur LDAP et Analyse de la configuration du service LDAP

Q120. Comment initialiser une nouvelle base et un nouveau contexte de nommage pour ce service d'annuaire ?

Réinitialiser la configuration du démon `slapd` avec le contexte de nommage demandé.

Reprendre les questions de la partie Réinitialisation de la base de l'annuaire LDAP

Q121. Comment activer la journalisation des transactions sur le service d'annuaire ?

Créer un fichier LDIF qui remplace la valeur par défaut de l'attribut `olcLogLevel` par `stats`.

Reprendre la question Comment activer la journalisation des manipulations sur les entrées de l'annuaire LDAP ?

Q122. Comment implanter les deux unités organisationnelles `people` et `groups` dans le nouvel annuaire ?

Créer un fichier LDIF qui décrit la création des deux unités organisationnelles dans le bon contexte. Ajouter ces deux unités organisationnelles dans l'annuaire.

Reprendre les questions Quelle est la syntaxe du fichier LDIF qui permet d'ajouter les deux unités organisationnelles (*organisational unit*) ? et Quelle est la commande à utiliser pour ajouter une ou plusieurs entrées dans l'annuaire ?

Q123. Comment implanter les quatre comptes utilisateurs dans cet annuaire ?

Créer un fichier LDIF qui décrit la création des des quatre comptes utilisateurs dans le bon contexte avec un jeu d'attributs complet pour l'authentification et le système de fichiers. Ajouter ces comptes dans l'annuaire.

Reprendre la question Quelle est la syntaxe du fichier LDIF qui permet d'ajouter les quatre utilisateurs avec leurs attributs système ?

2. Mise en œuvre de l'exportation NFS

Cette partie reprend les étapes décrites dans le support *Introduction au système de fichiers réseau NFSv4*. Après avoir traité la partie commune de la configuration NFS, il s'agit d'installer le paquet correspondant au serveur NFS et de créer l'arborescence des comptes utilisateurs à exporter avec le bon contexte de nommage.

2.1. Service NFS

Q124. Comment installer et valider les services commun au client et au serveur NFS ?

Rechercher et installer le paquet puis contrôler la liste des processus et des numéros de port ouverts.

On reprend ici les questions de la partie Gestion des paquets NFS

- Identification du paquet à installer.

```
apt search --names-only ^nfs- | egrep -v '(ganesh|^$)'

WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

En train de trier…
Recherche en texte intégral…
nfs-common/testing 1:2.6.3-3 amd64
  fichiers de prise en charge NFS communs au client et au serveur
  NFS server in User Space
nfs-kernel-server/testing 1:2.6.3-3 amd64
  gestion du serveur NFS du noyau
```

- Identification des processus actifs après installation du paquet.

```
ps aux | grep [r]pc
root      4876  0.0  0.0   6872  3180 ?        Ss   20:18   0:00 /sbin/rpcbind -f -w
```

Q125. Comment installer et configurer le paquet relatif à l'exportation d'une arborescence avec le protocole NFS ?

On reprend ici les questions de la partie Configuration du serveur NFS

- Identification du paquet à installer.

```
aptitude search '?and(nfs, server)'
p   nfs-kernel-server - gestion du serveur NFS du noyau
v   nfs-server
```

- Création de l'arborescence d'exportationNFS.

```
sudo mkdir -p /home/exports/home
```

- Ajout des instructions d'exportation dans le fichier de configuration du serveur NFS : `/etc/exports`.

```
cat << EOF | sudo tee -a /etc/exports
/home/exports           192.0.2.0/27(rw,sync,fsid=0,crossmnt,no_subtree_check)
/home/exports/home      192.0.2.0/27(rw,sync,no_subtree_check)
EOF
```

Q126. Comment valider la configuration de l'exportation réalisée par le serveur NFS ?

On reprend la question sur la la commande qui permet d'identifier l'arborescence disponible à l'exportation.

- Côté client, on utilise la commande showmountsuivie de l'option `-e` et de l'adresse IP du serveur à interroger.

- Côté serveur, on utilise la commande exportfs.

2.2. Montage local sur le serveur

Du point de vue métier, l'opérateur du réseau de stockage doit respecter le schéma de nommage qui veut que l'arborescence soit identique entre serveur et client. Dans ce but, réaliser un montage local permet de faire pointer l'arborescence partagée sur un volume de stockage donné. Ce volume de stockage pourra changer au cours du temps tout en respectant le schéma de nommage.

Q127. Quel est le montage local à mettre en place pour garantir la cohérence du schéma de nommage entre les postes serveur et client ?

On reprend ici la question sur la distinction entre les versions 3 et 4 du protocole NFS et sur le contexte de nommage.

- Création de la racine commune entre client et serveur.

```
sudo mkdir /ahome
```

- Montage local entre racine commune et arborescence exportée.

```
sudo mount --bind /home/exports/home /ahome
```

2.3. Création automatique du répertoire utilisateur

Cette étape correspond aux traitements réalisés lors de la toute première utilisation du service par un nouvel utilisateur.

Cette opération se déroule en plusieurs étapes dans la mesure où il est impossible de créer un répertoire utilisateur sur le serveur directement depuis le client.

1. Activer sur le serveur NFSv4 l'appel au module de création de répertoire utilisateur.
2. Toute les connexions suivantes depuis un client NFSv4 utiliseront l'arborescence utilisateur créée lors de la première connexion.

Avertissement

Cette opération suppose que l'on puisse utiliser le service LDAP sur le serveur lui-même. Il faut donc installer et configurer les paquets `libnss-ldapd`, `libpam-ldapd` sur le serveur de façon à accéder automatiquement aux ressources de l'annuaire.

Q128. Comment créer automatiquement l'arborescence d'un utilisateur qui n'existe que dans l'annuaire LDAP ?

Rechercher les fonctions de création automatique de répertoire utilisateur dans la liste des paquets de la distribution.

1. Sur le serveur, on ajoute le paquet oddjob-mkhomedir puis on complète le fichier commun de gestion de session : `/etc/pam.d/common-session`.

```
sudo pam-auth-update --package --enable mkhomedir
```

On peut vérifier le résultat en recherchant la clé `mkhomedir` dans les fichiers du répertoire `/etc/pam.d/`.

```
grep mkhomedir /etc/pam.d/*
```

2. Depuis un poste client différent du serveur, on provoque la création du répertoire utilisateur sur le serveur. Dans l'exemple ci-dessous, on utilise le service SSH pour déclencher la création du répertoire utilisateur ainsi que la copie des fichiers de paramétrage du *Shell*.

```
ssh padme@fe80::b8ad:caff:fefe:64%eth0
padme@fe80::b8ad:caff:fefe:64%eth0's password:
Creating home directory for padme.
Linux LDAP-Server 4.12.0-1-686-pae #1 SMP Debian 4.12.6-1 (2017-08-12) i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
padme@LDAP-Server:~$ pwd
/ahome/padme
```

Enfin, on lance une nouvelle connexion sur un client NFS de façon à tester l'automontage du répertoire utilisateur.

Avertissement

La connexion présentée ci-dessous n'est valide que si le service d'automontage fonctionne correctement. Il faut donc avoir traité la section suivante avant de faire ce test : Section 3, « Configuration de l'automontage avec le service LDAP ».

Sur un client avant connexion la liste des montage fait apparaître l'information suivante :

```
mount | grep ^ldap
ldap:ou=auto.home,ou=automount,dc=lab,dc=stri on /ahome type autofs \
(rw,relatime,fd=7,pgrp=12568,timeout=300,minproto=5,maxproto=5,indirect,pipe_ino=2875248)
```

```
etu@LDAP-Client:~$ su - padme
Mot de passe :
padme@LDAP-Client:/home/etu$ cd
padme@LDAP-Client:~$ pwd
/ahome/padme
padme@LDAP-Client:~$ mount | egrep '(ldap|nfs)'
ldap:ou=auto.home,ou=automount,dc=lab,dc=stri on /ahome type autofs \
  (rw,relatime,fd=7,pgrp=13510,timeout=300,minproto=5,maxproto=5,
  indirect,pipe_ino=2891149)
192.0.2.12:/home/padme on /ahome/padme type nfs4 \
  (rw,relatime,vers=4.2,rsize=131072,wsize=131072,
  namlen=255,hard,proto=tcp,timeo=600,
  retrans=2,sec=sys,clientaddr=192.0.2.25,local_lock=none,
  addr=192.0.2.12)
```

3. Configuration de l'automontage avec le service LDAP

Le principe de l'automontage veut que le montage d'une arborescence de système de fichiers réseau se fasse automatiquement et uniquement à l'utilisation. En effet, il n'est pas nécessaire de mobiliser les ressources du protocole NFS tant qu'une arborescence n'est pas effectivement parcourue. Dans le contexte de ce support, il n'est pas nécessaire de monter l'arborescence d'un répertoire utilisateur si celui-ci n'est pas connecté sur le poste client. On optimise ainsi les ressources du système et du réseau.

Du point de vue administration système, il est essentiel que la configuration des postes clients ne soit pas remise en question à chaque évolution du serveur ou à chaque ajout de nouveau compte utilisateur. C'est ici que le service LDAP intervient. Ce service sert à publier la configuration de l'automontage en direction des clients.

Pour appliquer ces principes, cette section doit couvrir les étapes suivantes.

- Pour compléter les informations publiées par le service LDAP, il faut ajouter un schéma spécifique à la fonction d'automontage et ensuite importer le contenu d'un fichier de description LDIF contenant les paramètres de configuration à diffuser vers les clients.

- Pour que le montage des arborescences soit automatique, il faut ajouter un paquet spécifique sur les systèmes clients et désigner le service LDAP comme fournisseur de la configuration. Cette désignation se fait à l'aide du *Name Service Switch*.

La principale difficulté dans le traitement des questions suivantes vient du fait qu'il est nécessaire d'échanger des informations entre le client et le serveur.

Dans le contexte de ce support, le service LDAP et le serveur NFS sont implantés sur le même système.

Q129. Quel est le paquet de la distribution Debian GNU/Linux qui fournit le service d'automontage via LDAP ?

Rechercher le mot clé ***automount*** dans le champ description du catalogue des paquets disponibles.

```
aptitude search "?description(automount)"
p   autodir                - crée automatiquement les répertoires home et
  group pour les comptes LDAP/NIS/SQL et locaux
p   autofs                 - montage automatique pour Linux basé sur le noyau
p   autofs-hesiod          - gestion de la carte Hesiod pour autofs
p   autofs-ldap            - gestion des schémas LDAP pour autofs
p   libnss-cache           - NSS module for using nsscache-generated files
p   libunix-configfile-perl - Perl interface to various Unix configuration files
p   ltspfsd                - Fuse based remote filesystem hooks for LTSP thin
  clients
p   nsscache               - asynchronously synchronise local NSS databases
  with remote directory services
p   vfu                    - Versatile text-based filemanager
```

Le paquet `autofs-ldap` correspond au besoin. On peut obtenir des informations supplémentaires en consultant sa description complète à l'aide de la commande `aptitude show autofs-ldap`.

Q130. Sur quel type de poste ce paquet doit il être installé ?

Le service d'automontage est a exécuter sur le poste qui ne détient pas le système de fichiers dans lequel se trouvent les répertoires utilisateur.

Ce paquet doit être installé sur le poste client puisque le processus `automount` doit être exécuté sur ce même client. Son installation se fait simplement avec la commande usuelle `sudo aptitude install autofs-ldap`.

Q131. Quelles sont les informations relatives au service LDAP à transférer entre client et serveur ?

Pour publier la configuration de l'automontage via le service LDAP, il est nécessaire de disposer du schéma de définition des attributs dans l'annuaire. Ce schéma est fourni avec le paquet `autofs-ldap` et doit être transféré vers le serveur LDAP pour compléter le catalogue des objets qu'il peut contenir.

```
dpkg -L autofs-ldap | grep schema
/etc/ldap/schema
/etc/ldap/schema/autofs.schema
```

```
cp /etc/ldap/schema/autofs.schema .
sed -i 's/caseExactMatch/caseExactIA5Match/g' autofs.schema
```

```
 scp autofs.schema etu@192.0.2.12:~
```

Au moment de la rédaction de ces lignes, le fichier de schéma livré avec le paquet `autofs-ldap` contient une erreur que l'on corrige à l'aide de la commande `sed`.

L'adresse IP utilisée dans la copie d'écran ci-dessus correspond au serveur LDAP et NFS.

Q132. Dans quel répertoire les informations transférées doivent elles être placées ?

Rechercher le répertoire de stockage des fichiers de schémas dans l'arborescence du serveur LDAP.

Une fois le fichier de schéma de transféré du client vers le serveur, celui-ci doit être placé dans l'arborescence du service LDAP avec les autres fichiers du même type.

```
sudo mv autofs.schema /etc/ldap/schema/
sudo chown root:root /etc/ldap/schema/autofs.schema
```

```
ls -lAh /etc/ldap/schema/autofs.schema
-rw-r--r-- 1 root root 830 sept. 27 10:29 /etc/ldap/schema/autofs.schema
```

Q133. Comment intégrer ces nouvelles informations d'automontage dans la configuration du service LDAP ?

L'intégration du nouveau schéma dans la configuration du serveur se fait en plusieurs étapes. Le fichier délivré avec le paquet `autofs-ldap` doit être converti en fichier LDIF avant d'être ajouté au DIT de configuration du démon `slapd`.

La conversion en fichier LDIF se fait à l'aide de la commande slaptest fournie avec le paquet `slapd`.

1. Création du répertoire de stockage du résultat de la conversion.

   ```
   mkdir schema-convert
   ```

2. Création du fichier de traitement des schémas. Comme de schéma `autofs` utilise des définitions issues des schémas de rang supérieur, il est nécessaire d'inclure les autres fichiers de schémas fournis avec le paquet.

   ```
   cat << EOF >schema-convert.conf
   include /etc/ldap/schema/core.schema
   include /etc/ldap/schema/cosine.schema
   include /etc/ldap/schema/inetorgperson.schema
   include /etc/ldap/schema/autofs.schema
   EOF
   ```

3. Conversion des fichiers de schémas au format LDIF.

   ```
   sudo slaptest -f schema-convert.conf -F schema-convert
   config file testing succeeded
   ```

4. Extraction des définitions utiles et formatage du résultat de la conversion. La commande ci-dessous élimine toutes les informations relatives à l'horodatage et à l'identification de l'utilisateur.

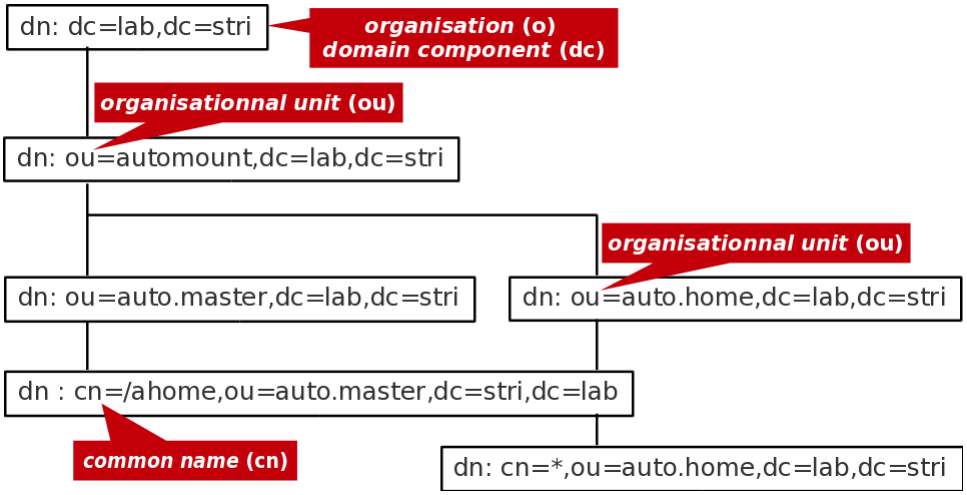
   ```
   cat schema-convert/cn\=config/cn\=schema/cn\=\{3\}autofs.ldif | \
   grep -Ev structuralObjectClass\|entryUUID\|creatorsName | \
   grep -Ev createTimestamp\|entryCSN\|modifiersName\|modifyTimestamp | \
   sed 's/dn: cn={.}autofs/dn: cn=autofs,cn=schema,cn=config/g' | \
   sed 's/{.}autofs/autofs/' > autofs.ldif
   ```

5. Ajout du schéma `autofs` dans la configuration du service.

   ```
   sudo ldapadd -Y EXTERNAL -H ldapi:/// -f autofs.ldif
   SASL/EXTERNAL authentication started
   SASL username: gidNumber=0+uidNumber=0,cn=peercred,cn=external,cn=auth
   SASL SSF: 0
   adding new entry "cn=autofs,cn=schema,cn=config"
   ```

Q134. Quelle est la syntaxe du fichier de description LDIF contenant la configuration de l'automontage ?

Le fichier de description ci-dessus correspond à l'arborescence suivante.



Arborescence LDAP de l'automontage - vue complète

```
cat ou-autofs.ldif
dn: ou=automount,dc=lab,dc=stri
ou: automount
objectClass: top
objectClass: organizationalUnit

dn: ou=auto.master,ou=automount,dc=lab,dc=stri
ou: auto.master
objectClass: top
objectClass: automountMap

dn: cn=/ahome,ou=auto.master,ou=automount,dc=lab,dc=stri
cn: /ahome
objectClass: top
objectClass: automount
automountInformation: ldap:ou=auto.home,ou=automount,dc=lab,dc=stri

dn: ou=auto.home,ou=automount,dc=lab,dc=stri
ou: auto.home
objectClass: top
objectClass: automountMap

dn: cn=*,ou=auto.home,ou=automount,dc=lab,dc=stri
cn: *
objectClass: top
objectClass: automount
automountInformation: -fstype=nfs4 192.0.2.12:/home/&
```

Q135. Comment intégrer ces définitions dans l'annuaire LDAP ?

Retrouver la syntaxe de la commande ldapadd qui permet d'insérer de nouvelles entrées dans l'annuaire.

On suit la même démarche que pour les comptes utilisateurs.

```
sudo ldapadd -cxWD cn=admin,dc=lab,dc=stri -f ou-autofs.ldif
Enter LDAP Password:
adding new entry "ou=automount,dc=lab,dc=stri"

adding new entry "ou=auto.master,ou=automount,dc=lab,dc=stri"

adding new entry "cn=/ahome,ou=auto.master,ou=automount,dc=lab,dc=stri"

adding new entry "ou=auto.home,ou=automount,dc=lab,dc=stri"

adding new entry "cn=*,ou=auto.home,ou=automount,dc=lab,dc=stri"
```

4. Accès aux ressources LDAP & NFS depuis le client

Dans cette section, on suppose que l'annuaire LDAP du poste serveur est complet et accessible. Dans un premier temps, on configure le poste client pour qu'il obtienne de façon transparente les informations sur les comptes utilisateurs desservis par l'annuaire. Dans un second temps, on complète sa configuration pour qu'il obtienne, toujours de façon transparente les informations sur le système de fichiers réseau.

Cette partie reprend les étapes décrites dans la section Configuration *Name Service Switch* du support *Introduction aux annuaires LDAP avec OpenLDAP*.

4.1. Configuration LDAP

Q136. Quels sont les paquets de bibliothèques LDAP relatifs au mécanisme *Name Service Switch* et au gestionnaire d'authentification PAM ?

Rechercher la liste des paquets dont le nom débute par `libnss` et `libpam`.

Les deux paquets utiles sont : `libnss-ldapd` et `libpam-ldapd`. Le paquet `nslcd` est une dépendance importante de `libnss-ldapd`. Il assure le volet connexion et authentification à l'annuaire.

Q137. Quelles sont les étapes de la configuration des paquets de bibliothèques NSS et PAM ?

Lors de l'installation des deux paquets, on passe par une série de menus debconf.

Voici un récapitulatif des réponses.

Pour le paquet `libnss-ldapd`, on donne la liste des services de nom à configurer :

- `passwd`
- `group`
- `shadow`

Pour le paquet `nslcd`, on donne les paramètres pour contacter le serveur LDAP.

- URI du serveur LDAP : ldap://192.0.2.12
- Base de recherche du serveur LDAP : dc=lab,dc=stri
- Authentification LDAP : aucune
- La base LDAP demande-t-elle une identification ? non
- Faut-il utiliser StartTLS ? non

Q138. Comment valider la configuration de l'accès à l'annuaire LDAP ?

Rechercher une commande permettant d'effectuer un appel système aux bibliothèques standard `libc`.

On qualifie le mécanisme *Name Service Switch* à l'aide de la commande getent.

```
getent passwd
root:x:0:0:root:/root:/bin/bash
<snip>
nslcd:x:111:117:nslcd name service LDAP connection daemon,,,:/var/run/nslcd/:/bin/false
padme:x:10000:10000:Padme Amidala Skywalker:/ahome/padme:/bin/bash
anakin:x:10001:10001:Anakin Skywalker:/ahome/anakin:/bin/bash
leia:x:10002:10002:Leia Organa:/ahome/leia:/bin/bash
luke:x:10003:10003:Luke Skywalker:/ahome/luke:/bin/bash
```

On qualifie l'authentication PAM à l'aide de la commande su.

```
su - luke
Mot de passe :
:/home/etu$
```

4.2. Configuration NFS avec automontage

On considère que le paquet `autofs-ldap` a déjà été installé pour fournir le schéma de la partie automontage au serveur LDAP. Voir Section 3, « Configuration de l'automontage avec le service LDAP ».

Q139. Quelle est la modification à apporter au fichier de configuration /etc/nsswitch.conf pour que le démon automount accède aux ressources de l'annuaire LDAP ?

Il faut ajouter une directive supplémentaire qui spécifie l'ordre de recherche des informations pour le démon `automount`.

La syntaxe est la suivante.

```
echo -e "\nautomount:      ldap" | sudo tee -a /etc/nsswitch.conf
```

Q140. Quel est le fichier de configuration du service d'automontage dans lequel sont définis ses paramètres globaux ?

Rechercher le répertoire dans lequel sont placés les fichiers de paramétrage de tous les services.

Il s'agit du fichier `/etc/default/autofs`.

Q141. Quelles sont les modifications à apporter à ce fichier pour que le démon accède à l'annuaire LDAP et que la journalisation soit active ?

Il faut éditer le fichier avec les éléments suivants.

- Désigner l'unité organisationnelle qui contient les entrées de configuration de l'automontage
- Faire apparaître les évènements du service d'automontage dans les journaux système
- Désigner le serveur LDAP à contacter
- Spécifier le point d'entrée pour les recherches dans l'annuaire

```
sudo grep -v ^# /etc/default/autofs
MASTER_MAP_NAME="ou=auto.master,ou=automount,dc=lab,dc=stri"
TIMEOUT=300
BROWSE_MODE="no"
LOGGING="verbose"
LDAP_URI="ldap://192.0.2.12"
SEARCH_BASE="ou=automount,dc=lab,dc=stri"
```

Q142. Comment vérifier que le service `autofs` a bien pris la nouvelle configuration en charge et fait appel aux ressources de l'annuaire LDAP ?

Rechercher dans les informations relatives au statut du service `autofs` les paramètres de configuration LDAP.

On affiche l'état du service à l'aide de la commande ci-dessous.

```
sudo systemctl status autofs
# autofs.service - Automounts filesystems on demand
   Loaded: loaded (/lib/systemd/system/autofs.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2017-09-15 13:58:18 CEST; 15s ago
  Process: 19416 ExecStart=/usr/sbin/automount $OPTIONS
    --pid-file /var/run/autofs.pid (code=exited, status=0/SUCCESS)
 Main PID: 19417 (automount)
    Tasks: 4 (limit: 4915)
   CGroup: /system.slice/autofs.service
           └─19417 /usr/sbin/automount --pid-file /var/run/autofs.pid

systemd[1]: Starting Automounts filesystems on demand...
automount[19417]: Starting automounter version 5.1.2,
   master map ou=auto.master,ou=automount,dc=lab,dc=stri
automount[19417]: using kernel protocol version 5.02
automount[19417]: connected to uri ldap://192.0.2.12
automount[19417]: mounted indirect on /ahome with timeout 300,
   freq 75 seconds
systemd[1]: Started Automounts filesystems on demand.
```

Q143. Quelles sont les méthodes qui permettent de valider le fonctionnement du service d'automontage ?

Donner deux moyens d'acquérir l'identité d'un utilisateur ou d'une utilisatrice défini(e) dans l'annuaire LDAP uniquement.

ne pas oublier le consulter les journaux système pour observer les étapes de ces connexions utilisateur.

- Connexion SSH depuis un autre hôte
- Changement d'identité sur le même hôte avec la commande su
- Utilisation du gestionnaire de connexion graphique

Enfin, une fois la session d'un(e) utilisat(eur|rice) défini dans l'annuaire LDAP ouverte, il est important de vérifier que l'automontage du répertoire personnel à fonctionné. Il suffit d'utiliser la commande mount pour afficher la liste des montages actifs.

On retrouve la copie d'écran donnée en fin de section précédente.

```
mount | egrep '(ldap|nfs)'
ldap:ou=auto.home,ou=automount,dc=lab,dc=stri on /ahome type autofs \
  (rw,relatime,fd=7,pgrp=875,timeout=300,minproto=5,maxproto=5,
  indirect,pipe_ino=16519)
192.0.2.12:/home/padme on /ahome/padme type nfs4 \
  (rw,relatime,vers=4.2,rsize=131072,wsize=131072,namlen=255,
  hard,proto=tcp,timeo=600,
  retrans=2,sec=sys,clientaddr=192.0.2.25,local_lock=none,
  addr=192.0.2.12)
```

5. Documents de référence

OpenLDAP Software 2.4 Administrator's Guide

Le guide *OpenLDAP Software 2.6 Administrator's Guide* est la référence essentielle sur le service LDAP.

Systèmes de fichiers réseau : NFS & CIFS

Systèmes de fichiers réseau : présentation des modes de fonctionnement des systèmes de fichiers réseau NFS & CIFS.

Linux NFS-HOWTO

Linux NFS-HOWTO : documentation historique complète sur la configuration d'un serveur et d'un client NFS jusqu'à la version 3 inclue.

Nfsv4 configuration

Nfsv4 configuration : traduction française extraite des pages du projet CITI de l'université du Michigan.