Annuaire LDAP - MDS

Debian GNU/Linux

Matthieu Vogelweith

17 août 2009

Résumé

Un annuaire électronique est une base de donnée spécialisée, dont la fonction première est de retourner un ou plusieurs attributs d'un objet grâce à des fonctions de recherche multi-critères. Contrairement à un SGBD, un annuaire est très performant en lecture mais l'est beaucoup moins en écriture. Sa fonction peut être de servir d'entrepôt pour centraliser des informations et les rendre disponibles, via le réseau à des applications, des systèmes d'exploitation ou des utilisateurs. Lightweight Directory Access Protocol (LDAP) est né de la nécessaire adaptation du protocole DAP (protocole d'accès au service d'annuaire X500 de l'OSI) à l'environnement TCP/IP. Initialement frontal d'accès à des annuaires X500, LDAP est devenu en 1995, un annuaire natif (standalone LDAP) sous l'impulsion d'une équipe de l'université du Michigan (logiciel U-M LDAP).

L'objectif de ce document est de détailler l'installation d'un serveur OpenLDAP [1] complet piloté par le MDS (**M**andriva **D**irectory **S**erver [2]) ainsi que les bases de son administration. Les différentes pages disponibles dans cette rubrique utiliseront dans la mesure du possible cet annuaire pour réaliser l'authentification des utilisateurs.

Ce document a été rédigé en LaTeX en utilisant l'excellent Vim sous Debian GNU/Linux. Il est disponible aux formats XHTML et PDF. Les sources LaTeX sont disponibles ici : LaTeX

Licence

Historique

- **17-08-2009** : Mise à jour pour Debian Lenny

Table des matières

Table des matières **4**

1 Serveur OpenLDAP **5**
1.1 Installation des paquets 5
1.2 Configuration du démon 5
1.3 Support SSL 6
1.4 Quelques tests 7

2 Gestion de l'annuaire : Mandriva Directory Server **9**
2.1 Présentation 9
2.2 Préparation 9
2.3 Installation de base 9
2.4 Configuration d'apache 10
2.5 NSS - Name Service Switch 11

3 Configuration des clients **13**
3.1 Authentification sur l'annuaire LDAP 13
3.1.1 PAM - Pluggable Authentication Module 13
3.2 Carnet d'adresses 14

4 Gestion des environnements complexe **15**

5 Replication LDAP avec SyncRepl **16**
5.1 Configuration du maître 16
5.2 Configuration des esclaves 16
5.3 Écriture depuis un esclave 17

6 Quelques outils **18**
6.1 slapcat 18
6.2 ldappasswd 18
6.3 ldapvi 18

7 Références **20**

Chapitre 1

Serveur OpenLDAP

1.1 Installation des paquets

L'implémentation LDAP la plus répandue actuellement dans le monde libre est OpenLDAP, disponible dans le paquet Debian "slapd". On peut également installer le paquet "ldap-utils" qui fournit quelques outils permettant de manipuler les annuaires.

```
# aptitude install slapd ldap-utils
```

Lors de la configuration de slapd, debconf demande simplement le mot de passe administrateur de l'annuaire. Il utilise ensuite le nom de domaine configuré sur la machine pour définir la racine de l'annuaire LDAP. Par exemple, si la machine locale s'appelle **ldap.example.org**, debconf va construire une arborescence LDAP ayant pour racine **dc=example,dc=org**.

Les fichiers de configurations se trouvent dans le répertoire "/etc/ldap/". On trouve, entre autre, le fichier **slapd.conf** qui est le fichier de configuration du démon et **ldap.conf** qui est le fichier de configuration pour les utilitaires installés précédemment avec le paquet ldap-utils.

La configuration initiale générée par debconf permet déjà de faire un premier test pour vérifier que le serveur fonctionne bien. On peut par exemple utiliser la commande **slapcat** qui va afficher le contenu de l'annuaire au format LDIF. Pour l'instant l'annuaire n'est pas très consistant, mais par la suite, il est préférable de re-diriger la sortie standard de slapcat dans less afin de pouvoir exploiter le résultat de la commande.

```
# slapcat | less
```

Notons que cette commande affiche un dump de l'annuaire sur la sortie standard en utilisant directement des fichiers de. Elle permet donc de visualiser le contenu de l'annuaire mais ne valide en aucun cas que le service fonctionne correctement. L'interrogation de l'annuaire en utilisant le protocole TCP dédié à cet effet avec la commande "ldapsearch" est décrite plus loin.

1.2 Configuration du démon

Avant de commencer à remplir et utiliser l'annuaire, il faut configurer une base et définir les droits d'accès. Toute cette configuration se fait dans **slapd.conf**. Ce fichier est divisé en plusieurs parties : d'abord une partie générale pour la configuration du démon puis une partie spécifique à chaque annuaire hébergé sur le serveur.

Dans un premier temps, il n'est pas nécessaire de faire de modifications dans la partie générale, debconf s'est chargé de faire une configuration minimale en fonction des réponses données lors de l'installation du paquet. On peut cependant examiner rapidement une section intéressante du fichier de configuration, la définition des ACLs :

```
access to attrs=userPassword
        by dn="cn=admin,dc=example,dc=org" write
        by anonymous auth
        by self write
        by * none

access to dn.base="" by * read

access to *
        by dn="cn=admin,dc=example,dc=org" write
        by * read
```

La configuration ci-dessus permet à l'administrateur d'éditer l'ensemble de l'annuaire alors que les utilisateurs anonymes peuvent simplement y accéder en lecture seule.

1.3 Support SSL

Par défaut, le démon **slapd** n'est actif qu'en mode "normal" sur le port 389. Pour permettre le support SSL, if faut bien sûr générer un certificat SSL qui permettra de s'assurer de l'authenticité du service et de chiffrer/déchiffrer les communications. Dans le cas du serveur LDAP, on peut créer le fichier de configuration **/etc/ssl/slapd.cnf** ci-dessous :

```
[ req ]
default_bits            = 2048
default_keyfile         = privkey.pem
distinguished_name      = req_distinguished_name
prompt                  = no
string_mask             = nombstr
x509_extensions         = server_cert

[ req_distinguished_name ]
countryName             = FR
stateOrProvinceName     = France
localityName            = Strasbourg
organizationName        = Example
organizationalUnitName  = LDAP Server
commonName              = ldap.example.org
emailAddress            = ldapmaster@example.org

[ server_cert ]
basicConstraints        = critical, CA:FALSE
subjectKeyIdentifier    = hash
keyUsage                = digitalSignature, keyEncipherment
extendedKeyUsage        = serverAuth, clientAuth
nsCertType              = server
nsComment               = "LDAP Certificate"
```

La génération du certificat se fait ensuite avec la commande suivante :

```
# openssl req -x509 -new \
    -config /etc/ssl/slapd.cnf \
    -out /etc/ssl/certs/slapd.pem \
    -keyout /etc/ssl/private/slapd.key \
    -days 730 -nodes -batch
```

Pour activer le support SSL de ce démon, il faut maintenant modifier la directive **SLAPD_SERVICES** du fichier /etc/default/slapd. Par exemple, pour que le démon soit disponible en mode normal et en mode SSL, il faut utiliser la configuration suivante : (Notons qu'en mode SSL, le démon écoute sur le port 636)

```
SLAPD_SERVICES="ldap:/// ldaps:///"
```

Il faut ensuite indiquer l'emplacement du certificat et de la clé SSL dans le fichier /etc/ldap/slapd.conf :

```
TLSCertificateFile /etc/ssl/certs/slapd.pem
TLSCertificateKeyFile /etc/ssl/private/slapd.key
```

Attention à bien positionner les droits pour les fichiers pour que le démon **slapd** puisse lire correctement la clé :

```
# chgrp openldap /etc/ssl/private/slapd.key
# chmod g+r /etc/ssl/private/slapd.key
```

Comme toujours, il faut maintenant redémarrer le service pour que les modifications soient prise en compte.

```
# /etc/init.d/slapd restart
```

On peut faire une rapide vérification en utilisant **nmap** : en principe la commande doit indiquer que le serveur écoute sur les ports 389 et 636 (ldap et ldapssl).

```
# nmap 127.0.0.1 -p 389,636
```

1.4 Quelques tests

L'annuaire est maintenant disponible sut le port 389 en TCP et de manière chiffrée sur le port 636 en TCP. Pour valider le bon fonctionnement du service, on peut interroger l'annuaire avec la commande **ldapsearch** fournie par le paquet **ldap-utils** installé précédemment. Par exemple, pour faire une requête LDAP :

```
# ldapsearch -x -H ldap://ldap.example.org -b "ou=Users,dc=example,dc=org" "(objectclass
    =*)"
```

Pour utiliser le mode SSL avec ldapsearch, il faut dans un premier temps ajouter la ligne suivante dans le fichier /etc/ldap/ldap.conf pour ignorer la vérification du certificat :

```
TLS_REQCERT allow
```

Il suffit maintenant de reprendre la même requête en modifiant l'URI et en ajoutant l'option -Z :

```
# ldapsearch -x -H ldaps://ldap.example.org:636 -Z -b "ou=Users,dc=example,dc=org" "(
    objectclass=*)"
```

Chapitre 2

Gestion de l'annuaire : Mandriva Directory Server

2.1 Présentation

La gestion d'un annuaire LDAP avec les outils fournis dans le paquets **ldap-utils** devient vite très complexe si l'annuaire nécessite des modifications régulières (ajout d'utilisateurs, de contacts, ...). Le MDS (**M**andriva **D**irectory **S**erver [2]) propose donc une solution ergonomique en AJAX permettant de gérer l'annuaire le plus simplement possible. Outre la simplicité d'administration, le MDS propose plusieurs plugins afin de gérer les services associés à l'annuaire comme Samba, Postfix Bind ou DHCPD.

Ce chapitre détaille donc l'installation de base du MDS, simplement pour permettre la gestion des utilisateurs des groupes. L'installation, la configuration et l'utilisation des différents plugins seront détaillés dans les pages dédiées aux différents services.

2.2 Préparation

Le MDS n'est pas encore disponibles des les dépôts officiels de Debian, il faut donc ajouter un dépôt spécifique dans /etc/apt/sources.list :

```
# Mandriva Directory Server
deb http://mds.mandriva.org/pub/mds/debian lenny main
```

Reste maintenant à mettre à jour la liste des paquets disponibles avec la commande suivante :

```
# aptitude update
```

2.3 Installation de base

```
# aptitude install mmc-agent mmc-web-base
```

- Création d'un hash en base64 du mot de passe administrateur de l'annuaire LDAP :

```
$ python -c 'print "{base64}"+"secret".encode("base64")'
{base64}c2VjcmV0
```

- Configuration de l'accès à l'annuaire LDAP dans /etc/mmc/plugins/base.ini

```
baseDN = dc=example, dc=org
password = <mot_de_passe_admin_ldap_en_base64>

passwordscheme = ssha
```

Dans ce fichier de configuration, il est également possible de définir les attributs qui seront ajoutés automatiquement à la création d'un utilisateur. Par exemple, pour que les utilisateurs aient, par défaut, le droit de modifier leur mot de passe, ajouter la section suivante à la fin du fichier /etc/mmc/plugins/base.ini :

```
[userdefault]
objectClass = +lmcUserObject
lmcACL = :base#users#passwd/
```

- Configuration de l'interface web dans /etc/mmc/mmc.ini.

```
root = /
```

La configuration de base de la MMC étant effectuée, il faut maintenant enrichir l'annuaire LDAP avec la schéma de la MMC. Ce schéma va simplement fournir quelques attributs supplémentaire à l'annuaire pour la gestion interne de la MMC (les droits, ...). Pour ajouter le schéma, il faut dans un premier temps le copier dans le répertoire /etc/ldap/schema/ :

```
# cp /usr/share/doc/python-mmc-base/contrib/ldap/mmc.schema /etc/ldap/schema/
```

Il faut ensuite configurer slapd pour qu'il utilise ce nouveau schéma en ajoutant la ligne suivante dans /etc/ldap/slapd.conf :

```
include         /etc/ldap/schema/mmc.schema
```

Et enfin redémarrer les services pour que les modifications soient prises en compte :

```
# /etc/init.d/slapd restart
# /etc/init.d/mmc-agent restart
```

2.4 Configuration d'apache

Ecoute sur le port 443 dans le fichier /etc/apache2/ports.conf :

```
Listen 443
```

Suppression de l'alias dans httpd.conf et ajout d'un vhost dans le fichier /etc/apache2/sites-available/02_mmc :

```
<VirtualHost *:80>

    ServerName mmc.example.org
    ServerAlias mmc
    ServerAdmin webmaster@example.org

    RewriteEngine on
    RewriteCond %{SERVER_PORT} !^443$
    RewriteRule ^/(.*)$ https://%{SERVER_NAME}%{REQUEST_URI} [R=301,L]

    ErrorLog /var/log/apache2/mmc_error.log
    CustomLog /var/log/apache2/mmc_access.log combined

</VirtualHost>

<VirtualHost *:443>

    ServerName mmc.example.org
    ServerAlias mmc
    ServerAdmin webmaster@example.org

    SSLEngine on
    SSLCertificateFile /etc/ssl/certs/httpd.pem
    SSLCertificateKeyFile /etc/ssl/private/httpd.key

    DocumentRoot /usr/share/mmc

    <Directory /usr/share/mmc>
        AllowOverride None
        Order allow,deny
        allow from all
        php_flag short_open_tag on
    </Directory>

    ErrorLog /var/log/apache2/mmc_error.log
    CustomLog /var/log/apache2/mmc_access.log combined

</VirtualHost>
```

Activation du nouveau vhost et redémarrage du service :

```
# a2ensite 02_mmc
# /etc/init.d/apache2 restart
```

2.5 NSS - Name Service Switch

```
# aptitude install libnss-ldap
```

En premier lieu, on indique à NSS qu'il faut chercher les utilisateurs dans l'annuaire LDAP. Pour cela, il suffit d'éditer le fichier **/etc/nsswitch.conf** et de modifier les trois lignes ci-dessous. Notons qu'avec la configuration suivante, le système va d'abord utiliser les listes d'utilisateurs locales (/etc/-passwd, /etc/group, ...) avant de chercher dans l'annuaire. Ceci permet d'utiliser les deux types de comptes simultanément et permet d'éviter les problèmes en cas d'indisponibilité du serveur LDAP.

```
passwd:         files ldap
group:          files ldap
shadow:         files ldap
```

Maintenant que NSS sait qu'il faut utiliser LDAP pour chercher les utilisateurs qui ne sont pas déclarer localement, il faut configurer le module libnss-ldap pour lui donner les paramètres du serveur LDAP à utiliser. Si vous avez répondu correctement aux questions de debconf, la configuration doit être opérationnelle. On peut cependant vérifier les trois lignes suivantes dans le fichier **/etc/libnss-ldap.conf** :

```
host 127.0.0.1
base dc=example,dc=org
ldap_version 3
```

Si tout c'est passé correctement, on peut maintenant consulter les informations d'un utilisateur LDAP en exécutant la commande suivante :

```
# getent passwd matthieu
matthieu:x:1000:1000:Matthieu Vogelweith:/home/matthieu/:/bin/bash
```

Pour information, le paquet **nscd** a peut-être été installé en même temps que la librairie libnss-ldap. Ce paquet permet de mettre en cache les requêtes de NSS vers LDAP afin de minimiser la charge de l'annuaire. En conséquence, si de nombreuses modifications sont faites pendant la période de test du serveur, les résultats peuvent être erronés. Sur un système en production, notamment si Samba utilise l'annuaire, il est fortement conseillé de supprimer ce paquet afin de ne pas obtenir des comportements inexplicables lors de l'accès à l'annuaire.

Chapitre 3

Configuration des clients

Maintenant que l'annuaire est en place et qu'il contient quelques utilisateurs, il faut configurer les clients qui souhaitent utiliser cet annuaire. Si l'annuaire a bien été conçu, il peut être utilisé pour beaucoup de fonction différentes. La première étant l'authentification. L'avantage de LDAP est qu'on peut l'utiliser comme support d'authentification pour un grand nombre de services et avoir ainsi une base des utilisateurs centralisée.

3.1 Authentification sur l'annuaire LDAP

Actuellement, la plupart des applications utilisent une librairie pour obtenir des informations sur les utilisateurs et une autre pour réaliser leurs authentifications. La première tâche est confiée à NSS (**N**ame **S**ervice **S**witch) qui se charge de fournir une liste des utilisateurs, des groupes, etc. L'authentification est par contre confiée à PAM (**P**luggable **A**uthentication **M**odule) qui va alors vérifier la validité des identifiants et proposer différentes actions a exécuter lorsqu'un utilisateur s'authentifie.

Pour authentifier un utilisateur sur un annuaire LDAP, il faut donc ajouter le support LDAP à NSS et à PAM. Ceci est réalisé grâce aux deux librairies *libnss-ldap* et *libpam-ldap*.

Le système est maintenant capable de chercher des utilisateurs dans l'annuaire, il ne reste plus qu'a les authentifier par l'intermédiaire de PAM.

3.1.1 PAM - Pluggable Authentication Module

```
# aptitude install libpam-ldap
```

Comme pour la librairie *libnss-ldap*, la configuration est normalement opérationnelle suite aux questions de debconf mais on peut toujours vérifier les éléments suivants dans le fichier /**etc/-pam_ldap.conf** :

```
host 127.0.0.1
base dc=example,dc=org
ldap_version 3
rootbinddn cn=admin,dc=example,dc=org
pam_password crypt
```

- configuration de /etc/pam.d/common-account

```
account         sufficient      pam_ldap.so
account         required        pam_unix.so try_first_pass
```

- configuration de /etc/pam.d/common-auth

```
auth            sufficient      pam_ldap.so
auth            required        pam_unix.so nullok_secure try_first_pass
```

- configuration de /etc/pam.d/common-password

```
password        sufficient      pam_ldap.so
password        required        pam_unix.so nullok obscure min=4 max=8 md5 try_first_pass
```

En principe, il est maintenant possible de s'authentifier correctement sur une machine cliente en utilisant un des utilisateurs créés dans l'annuaire. Il est également possible de modifier le mot de passe de chacun en utilisant la commande bien connue **password** : le mot de passe sera mis à jour directement dans l'annuaire.

Il manque maintenant une chose essentielle : les répertoires des utilisateurs. En effet lors de la création des utilisateurs dans l'annuaire, on spécifie un chemin de **homeDir** mais a aucun moment ce répertoire n'a été créé : c'est l'objet de la section suivante.

3.2 Carnet d'adresses

On peut maintenant utiliser cette base comme annuaire au premier sens du terme, pour les clients mails par exemple. En effet il est maintenant très simple de demander aux clients mails de remplir le carnet d'adresse avec l'annuaire LDAP. Avec Thunderbird, par exemple, il faut ouvrir le carnet d'adresse puis ajouter un répertoire LDAP (Menu Fichier->Nouveau). On peut télécharger l'annuaire complet ou rester en mode connecté.

Chapitre 4

Gestion des environnements complexe

- Utilisation de plusieurs agents
- Gestion des réseaux multi-sites avec le MDS

Chapitre 5

Replication LDAP avec SyncRepl

5.1 Configuration du maître

Dans /etc/ldap/slapd.conf

```
# rootdn directive for specifying a superuser on the database. This is needed
# for syncrepl.
rootdn          "cn=admin,dc=example,dc=org"

moduleload syncprov
sessionlog   123 500
overlay syncprov
syncprov-checkpoint 50 5
syncprov-sessionlog 50
```

Reste enfin à redémarrer le service sur l'annuaire maître pour que la réplication fonctionne.

```
# /etc/init.d/slapd restart
```

5.2 Configuration des esclaves

Coté esclave, il est existe deux méthodes de réplication lorsqu'on utilise SyncRepl :

- **refreshOnly** : L'esclave se met à jour périodiquement en utilisant les données du maître. Par exemple, il se connecte toute les heures pour vérifier si des données ont été modifiées.
- **refreshAndPersist** : L'esclave conserve une connexion persistante avec l'annuaire maître et les données sont répliquées en temps réel lors de chaque modification sur le maître.

L'exemple ci-dessous décrit la configuration d'un esclave en mode "refreshAndPersist" qui présente beaucoup plus d'intérêt dans le sens ou ce mode permet d'obtenir une synchro en temps réél de l'annaire esclave.

```
index entryUUID,entryCSN eq
rootdn          "cn=admin,dc=example,dc=org"

syncrepl rid=421
    provider=ldap://ip_du_serveur_LDAP_maitre:389
    type=refreshAndPersist
    retry="60 10 300 +"
    searchbase="dc=example,dc=org"
    schemachecking=on
```

```
    bindmethod=simple
    binddn="cn=admin,dc=example,dc=org"
    credentials=mon_mot_de_passe
```

Cette configuration impose la vérification des schémas grâce à l'option **schemachecking=on**. Il faut donc impérativement que l'ensemble des schémas utilisés sur le maître soient disponible sur les esclaves. Une méthode rapide peut-être de copier simplement tous les schémas du maître avec une commande du type :

```
# scp root@ip_du_serveur_LDAP_maitre:/etc/ldap/schema/*.schema /etc/ldap/schema/
```

Bien évidemment, il reste à modifier le fichier slapd.conf pour correspondre à la configuration du maître. Il faut notamment modifier les paramètres suivants :

- La liste des schémas utilisés (include /etc/ldap/schema/nom_du_schema)
- Le suffix de l'annuaire (par exemple "dc=example,dc=org")
- Les ACLs si le DN de l'administrateur LDAP est différent.

Contrairement à slurpd, avec SyncRepl l'annuaire esclave ne doit pas nécessairement être identique à l'annuaire maître lorsque l'on démarre la réplication. La méthode la plus simple est donc de supprimer complètement les données existantes sur l'esclave et de démarrer ensuite la réplication. De cette façon, au démarrage du démon, syncrepl va copier l'intégralité de l'annuaire maître sur l'esclave.

```
# /etc/init.d/slapd stop
# rm /var/lib/ldap/*
# /etc/init.d/slapd start
```

Avec cette configuration, le client établi une connexion persistante vers l'annuaire maître et synchronise l'annuaire en temps réel. Il est possible de vérifier que la connexion est bien établie avec la commande netstat :

```
# netstat -naptu | grep "ESTABLISHED.*slapd"
```

Le résultat de cette commande doit montrer une connexion TCP établie entre l'adresse IP de l'esclave et l'adresse IP du maître sur le port 389.

5.3 Écriture depuis un esclave

La réplication proposée par OpenLDAP < 2.4.0 est une réplication de type maître / esclave. C'est à dire qu'il n'est possible de faire des modifications que sur le maître. Pour contourner le problème, il est possible d'indiquer à un esclave de transmettre les demandes d'écriture à l'annuaire maître. Pour cela, il suffit d'ajouter la ligne suivante dans le fichier /etc/ldap/slapd.conf de l'esclave :

```
updateref ldap://ip_du_serveur_LDAP_maitre:389
```

Chapitre 6

Quelques outils

6.1 slapcat

L'utilitaire slapcat déjà été utilisé pour vérifier le contenu de l'annuaire dans cette page. Il peut également servir à générer un dump de l'annuaire pour réaliser des sauvegardes par exemple. La commande ci-dessous génère par exemple undump de l'annuaire et l'enregistre dans un fichier daté :

```
# slapcat > `date +ldap_dump_%Y%m%d_%H%M%S`
```

6.2 ldappasswd

Le mot de passe de l'administrateur LDAP renseigné lors de la configuration du paquet **slapd** est toujours modifiable en utilisant la commande suivante (en remplaçant "new_passwd" et "old_passwd" par les valeurs désirées) :

```
# ldappasswd -s new_passwd -D "cn=admin,dc=example,dc=org" -w old_passwd -x cn=admin,dc=
    example,dc=org
```

Notons que la commande **ldappasswd** est fournie pour le paquet **ldap-utils**.

6.3 ldapvi

Pour les adeptes de l'éditeur VI, il existe un outils très intéressant permettant d'éditer le contenu d'un annuaire LDAP avec VI : **ldapvi**. Il est disponible dans les dépôts officiels Debian dans le paquet du même nom :

```
# aptitude install ldapvi
```

Il est alors possible d'éditer l'annuaire sans être obliger de générer un fichier LDIF et d'utiliser les commandes ldapmodify ou ldapadd. Par exemple, pour éditer un annuaire avec un accès total, utilisez la commande suivante :

```
# ldapvi --discover --host xxx.xxx.xxx.xxx --user "cn=admin,dc=example,dc=org" --password
    "ldap_password"
```

Pour appliquer les modifications, il suffit de quitter VI avec la commande habituelle : **ESC :wq**. Notons que ldapvi peut être utiliser avec n'importe quel éditeur de texte, il suffit de le définir dans le variable d'environnement EDITOR.

Chapitre 7

Références

[1] Site officiel de openldap. www.openldap.org.

[2] Site communautaire du mandriva directory server. mds.mandriva.org.

[3] Site officiel du projet debian. www.debian.org.