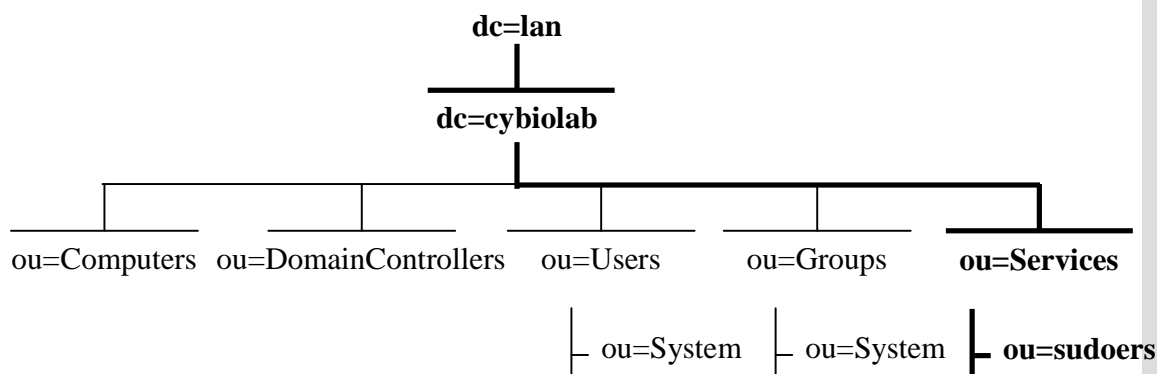




## Arborescence de SUDO dans LDAP

Nous allons ajouter une branche dans notre arborescence LDAP afin d'y emmagasiner nos règles sudo. En procédant ainsi, il sera plus facile d'en faire la gestion et aussi d'y appliquer des ACL sur l'ensemble de notre annuaire LDAP.



## Ajustement du DIT pour le service SUDO

Nous allons ajouter une branche à l'arborescence existante dans notre annuaire LDAP.

Créer un fichier portant le nom `sudoer.ldif` et inscrivez-y les lignes suivantes.

**vim sudoer.ldif**

```

dn: ou=Services,dc=cybiolab,dc=lan
ou: Services
objectClass: top
objectClass: organizationalUnit
description: Default container for services

dn: ou=Sudoers,ou=Services,dc=cybiolab,dc=lan
ou: Sudoers
objectClass: top
objectClass: organizationalUnit
  
```

Exécuter le script bash que nous avons utilisé précédemment au module 5 unité 2.

**./import\_ldif.sh sudoer.ldif**

NOTES



## Configuration du service LDAP pour SUDO

Vous devez étendre maintenant votre schéma LDAP pour que ce dernier supporte les attributs de sudo.

Dans le répertoire `/etc/openldap/schema`, créer un fichier et appeler le `sudo.schema`.

**vim /etc/openldap/schema/sudo.schema**

```
# schema file for sudo

attributetype ( 1.3.6.1.4.1.15953.9.1.1
  NAME 'sudoUser'
  DESC 'User(s) who may run sudo'
  EQUALITY caseExactIA5Match
  SUBSTR caseExactIA5SubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

attributetype ( 1.3.6.1.4.1.15953.9.1.2
  NAME 'sudoHost'
  DESC 'Host(s) who may run sudo'
  EQUALITY caseExactIA5Match
  SUBSTR caseExactIA5SubstringsMatch
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

attributetype ( 1.3.6.1.4.1.15953.9.1.3
  NAME 'sudoCommand'
  DESC 'Command(s) to be executed by sudo'
  EQUALITY caseExactIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

attributetype ( 1.3.6.1.4.1.15953.9.1.4
  NAME 'sudoRunAs'
  DESC 'User(s) impersonated by sudo'
  EQUALITY caseExactIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

attributetype ( 1.3.6.1.4.1.15953.9.1.5
  NAME 'sudoOption'
  DESC 'Options(s) followed by sudo'
  EQUALITY caseExactIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

attributetype ( 1.3.6.1.4.1.15953.9.1.6
  NAME 'sudoRunAsUser'
  DESC 'User(s) impersonated by sudo'
  EQUALITY caseExactIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )
```

NOTES



```

attributetype ( 1.3.6.1.4.1.15953.9.1.7
  NAME 'sudoRunAsGroup'
  DESC 'Group(s) impersonated by sudo'
  EQUALITY caseExactIA5Match
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.26 )

objectclass ( 1.3.6.1.4.1.15953.9.2.1 NAME 'sudoRole' SUP top STRUCTURAL
  DESC 'Sudoer Entries'
  MUST ( cn )
  MAY ( sudoUser $ sudoHost $ sudoCommand $ sudoRunAs $
    sudoRunAsUser $ sudoRunAsGroup $ sudoOption $ description )
)

```

Ajouter la ligne suivante dans votre fichier *slapd.conf*, immédiatement après les autres inclusions.

**vim /etc/openldap/slapd.conf**

```
include /etc/openldap/schema/sudo.schema
```

Ainsi que la ligne d'indexation à la toute fin du fichier *slapd.conf*. Ceci va permettre de rendre l'utilisation du sudo avec ldap plus efficace.

```
index sudoUser eq
```

Vérifier votre configuration avec *slaptest* et redémarrer votre serveur OpenLDAP afin qu'il prenne en compte les modifications.

**slaptest**

```
config file testing succeeded
```

**/etc/init.d/slaped restart**

Faites ensuite la modification des fichiers *ldap.conf* sur tous les postes qui utiliseront le SUDO par LDAP. Ajouter au fichier l'entrée suivante à la toute fin du fichier.

**vim /etc/ldap.conf**

```
sudoers_base ou=Sudoers,ou=Services,dc=cybiolab,dc=lan
```

NOTES



A partir d'ici, nous allons configurer la section cliente du serveur pour que lui-même puisse bénéficier du sudo avec LDAP. Il nous suffit de configurer l'authentification par NSS ainsi qu'un ajout à la section cliente de LDAP.

NOTES



### Configuration de NSS pour le support LDAP de SUDO

Procédons à la modification du fichier *nsswitch.conf* en ajoutant le paramètre suivant.

**vim /etc/nsswitch.conf**

```
...
sudoers :          ldap files
```

### Configuration de LDAP pour le support LDAP de SUDO

Dernière configuration côté client du serveur qui consiste à modifier le fichier *ldap.conf.sudo*.

**vim /etc/ldap.conf.sudo**

```
uri ldap://serveurpdc.cybiolab.lan
ldap_version      3
sudoers_base      ou=Sudoers,ou=Services,dc=cybiolab,dc=lan

# sudoers_debug 2
```

### sudoers\_debug

Ce paramètre permet d'activer l'audit d'utilisation du sudo.

Valeur	Description
1	Retourne des informations limitées de dépannage.
2	Affiche la totalité des correspondances d'informations de dépannage.

Syntaxe:

```
sudoers_debug=2
```

## Utilisation de SUDO avec LDAP

### Création des règles par défaut du SUDO

Nous allons créer le fichier LDIF qui contiendra la règle Sudo par défaut ainsi que la règle Sudo pour l'utilisateur root du système.

**vim default\_rules.ldif**

```
dn: cn=defaults,ou=Sudoers,ou=Services,dc=cybiolab,dc=lan
objectClass: top
objectClass: sudoRole
cn: defaults
description: Default sudo rule
sudoOption: env_keep+=SSH_AUTH_SOCK
sudoOption: timestamp_timeout=0
```

```
dn: cn=root,ou=Sudoers,ou=Services,dc=cybiolab,dc=lan
objectClass: top
objectClass: sudoRole
cn: root
sudoUser: root
sudoHost: ALL
sudoRunAsUser: ALL
sudoCommand: ALL
```

Nous allons donc faire l'importation des nouvelles entrées dans l'annuaire LDAP.

```
ldapadd -x -f "/tmp/sudoers.ldif" -D "cn=ldapadmin,dc=cybiolab,dc=lan" -W
```

### Création de nouvelles règles

La création des règles sudo avec LDAP peuvent se faire à l'aide de l'outil web phpLDAPAdmin. Pour cela vous devez utiliser le modèle de création CySudo que vous copiez dans le répertoire `/var/www/phpldapadmin/templates`. Peser sur le bouton pour vider le cache de phpLDAPAdmin.

Voici un exemple que nous vous conseillons d'appliquer sur votre contrôleur de domaine Linux.

```
dn: cn=%DomainAdmins=Sudoers,ou=Services,dc=cybiolab,dc=lan
objectClass: top
objectClass: sudoRole
```

NOTES



```
cn: DomainAdmins
sudoUser: %DomainAdmins
sudoHost: ALL
sudoRunAsUser: ALL
sudoCommand: ALL
```

Cette règle indique que le groupe *DomainAdmins* a le droit d'exécuter toute les commandes sur tous les ordinateurs sous l'utilisateur de son choix.

Si elle était appliquée localement sur un poste Linux elle ressemblerait à ceci:

```
%DomainAdmins ALL=(ALL) ALL
```

### Création d'une règles sudo avec LDAP

1. Ouvrez l'outil web phpLDAPAdmin.
2. Dans le volet de gauche, cliquer sur le plus de l'unité organisationnel Services, puis Sudoers.
3. Puis cliquer sur "*Create new entry here*".
4. Le modèle de création qui nous intéresse est "*Domain Sudo Policy*".
5. Vous devez maintenant entrer des valeurs aux attributs pour la création de votre règle sudo.

Le champ **Container DN** vous indique l'emplacement où sera situé votre groupe dans l'arborescence LDAP. Cet entré devrait pointer sur l'unité d'organisation *ou=Sudoers,ou=Services*.

Entrons ensuite le **Common Name**. Ne mettez pas d'espace lors de l'inscription du nom. L'outil phpLDAPAdmin et l'annuaire LDAP supporte les espaces, mais cela peut vous causer des ennuis pour les autorisations ou pour la gestion en ligne de commande.

Le champ **Description** est simplement à titre indicatif. Par défaut, le modèle affiche "*Domain Sudo Rule*".

Le champ **sudoCommand** contient une ou plusieurs commandes autorisées par l'administrateur. Il vous est possible d'ajouter plus de commande que ce que le modèle de création propose.

L'utilisation de l'exclamation (!) en préfixe, indique que la commande qui suit est interdite à l'utilisateur ou au groupe.

NOTES



Vous devez cependant les ajouter après la création de l'objet LDAP. Vous pouvez utiliser ALL pour indiquer que toutes les commandes peuvent être sudoer sans exception, mais gardez vous une gêne de le faire.

Le champ **sudoRunAsUser** permet d'entrer un nom de l'utilisateur sous laquelle cette règle autorise l'exécution des commandes du sudo. Vous pouvez utiliser ALL pour indiquer tout les utilisateurs sans exception.

Contrairement au sudo local, vous ne pouvez pas mettre d'exclusion (!) sur ce champ.

Le champ **sudoRunAsGroup** permet d'entrer un nom du groupe sous laquelle cette règle autorise l'exécution des commandes du sudo. Vous pouvez utiliser ALL pour indiquer tout les groupes sans exception.

Contrairement au sudo local, vous ne pouvez pas mettre d'exclusion (!) sur ce champ.



**NOTE:** Nous le redirons jamais assez, attribuer toujours vos permissions au niveau des groupes et jamais au niveau des utilisateurs.

Dans le champ **sudoHost**, ajouter le nom des ordinateurs, adresses IP ou de sous-réseaux qui seront autorisés dans l'application du sudo.

Contrairement au sudo local, vous ne pouvez pas mettre d'exclusion sur ce champ.

Le champ **sudoOption** peut contenir l'option *!authenticate* par exemple. Cette option, appeler NOPASSWD dans le fichier sudoers, est intéressante pour que l'utilisateur de cette règle n'ait pas à rentrer le mot de passe de l'utilisateur sous laquelle la commande sera lancée.

Le champ **sudoUser** contient les utilisateurs ou groupes ayant les droits d'utiliser les commandes spécifiées par cette règle. Pour les utilisateurs, inscrivez simplement le nom de ce dernier, pour les groupes ajouter un signe % avant le nom du groupe désiré. Par exemple: *%DomainAdmins*. Encore ici, contrairement au sudo local, vous ne pouvez pas mettre d'exclusion sur ce champ.

## NOTES





## logfile

Ce paramètre définit le fichier dans lequel inscrire les utilisations de la commande sudo sur le serveur ou la commande sera lancée.

Syntaxe:

```
logfile=/var/log/sudo
```

## ignore\_local\_sudoers

Ce paramètre désactive complètement l'utilisation du fichier sudoers sur le serveur. S'applique uniquement dans l'entrée *defaults* de LDAP, soit *cn=defaults,ou=Sudoers,ou=Services,dc=cybiolab,dc=lan*.

Syntaxe:

```
ignore_local_sudoers
```

## Différences entre le SUDO et SUDO avec LDAP

Les différences majeures entre le sudo local et le sudo avec ldap

- L'emplacement de définition des règles sudo. L'un est dans le fichier sudoers, alors que l'autre se retrouve dans l'annuaire LDAP.
- L'application d'exclusion (négation) ne s'applique pas sur les attributs suivant, *sudoHost*, *sudoUser*, *sudoRunAsUser* et *sudoRunAsGroup* pour le sudo avec LDAP.
- La portée d'application des règles sudo qui sont limité qu'à un seul poste Linux pour le sudo local, et à l'ensemble du domaine pour le sudo avec LDAP.
- Les options du sudo local sont remplacées dans le sudo avec LDAP.

NOPASSWD	!authenticate
PASSWD	authenticate
NOEXEC	noexec
EXEC	!noexec

- L'utilisation de la commande visudo ne s'applique pas pour les règles contenues dans l'annuaire LDAP.

NOTES



## Règles de délégation du domaine

Nous vous proposons d'ajouter ces règles sudo supplémentaires pour faire la délégation de contrôle dans votre domaine Linux.

La première règle sudo qui suit va permettre au groupe *DomainAdmins* d'avoir plein contrôle sur tous les ordinateurs du domaine fonctionnant sous Linux.

```
dn: cn=%DomainAdmins,ou=Sudoers,ou=Services,dc=cybiolab,dc=lan
objectClass: top
objectClass: sudoRole
cn: %DomainAdmins
sudoUser: %DomainAdmins
sudoHost: ALL
sudoCommand: ALL
sudoRunAsUser: ALL
description: Domain Sudo Rule
```

La seconde règle sudo qui suit va permettre au groupe *PrintOperators* de faire la gestion du service cups sur tous les ordinateurs du domaine fonctionnant sous Linux. Cette règle est par contre relativement permissive. A vous de juger sa pertinence sur votre domaine. A noter que nous n'avons pas encore vu l'installation du service cups.

```
dn: cn=%PrintOperators,ou=Sudoers,ou=Services,dc=cybiolab,dc=lan
objectClass: top
objectClass: sudoRole
cn: %PrintOperators
sudoUser: %PrintOperators
sudoHost: ALL
sudoCommand: /etc/init.d/cupsd
sudoCommand: sudoedit /etc/cups/cupsd.conf
sudoRunAsUser: ALL
description: Domain Sudo Rule
```

NOTES

