LDAP Parte 2 – Autenticando Usuários Ldap no Linux

By admin on 6 de fevereiro de 2012 in News

Na primeira parte deste artigo, LDAP Parte 1 – Introdução ao LDAP, vimos como instalar e configurar nosso LDAP.

Falamos um pouco sobre os Schemas, mas não abordamos de fato o que são e onde estão.

Para facilitar um pouco, veremos mais um pouco sobre Schemas.

O esquema ou Schema é uma coleção de classes de objetos e seus atributos. Fácil não?!!

Está bem, não tão fácil… Vamos ver o que são essas tais de classes de objetos.

Uma classe de objeto define quais entradas são válidas dentro de um sistema LDAP. Por exemplo, se queremos cadastrar um usuário podemos chamar uma classe de objeto chamada user (fictício), e dentro dessa classe serão definidos os atributos, como nome, endereço, telefone, entre outras coisas.

O schema traz uma coleção de objetos, para que possam ser usados por determinados sistemas, como por exemplo, podemos ter um Schema para o Samba, NFS, e no nosso caso contas Posix do Linux.

A classe de objetos que usaremos será **posixAccount, shadowAccount** para integrar o nosso Linux com Ldap.

Usaremos também, a configuração do cliente LDAP, configurando diretamente os arquivos de autenticação do PAM (Plugable Authentication Module), que é o sistema padrão de autenticação do Linux.

Vimos no artigo anterior, o uso dos objetos **top, organizationalUnit, person, e posixGroup.** Neste artigo ainda usaremos a base vista anteriormente, mas ao final do artigo, estaremos já autenticando usando uma base LDAP.

Considerações Iniciais

O Ambiente usado neste artigo é o mesmo do artigo anterior, então temos o domínio **“dominiolinux.net”.**

Caso, o seu ambiente não esteja configurado, basta verificar o artigo anterior para criar todo ambiente.

Estamos usando neste caso, o Debian 5.0 (Lenny) e OpenLdap versão 2.4.11.

Falando da Configuração

Já é o momento de falarmos um pouco do servidor LDAP. Seu arquivo de configuração está em **/etc/ldap/slapd.conf.** Neste arquivo teremos a configuração inicial feita anteriormente, dos mais importante temos:

- **include** Está opção inclui um arquivo externo, que será lido durante a inicialização do serviço. Normalmente utilizado para incluir os arquivos do Schema. Podemos ver em nosso arquivo que o ele faz referencia para os arquivos: core.schema, cosine.schema, nis.schema e inetorgperson.schema, que são os schemas utilizados pelo Sistema.
- **pidfile** Informa onde será gravado o arquivo que conterá o PID do processo gerado pelo slapd.
- **argsfile** Argumentos passador ao Servidor.
- **loglevel** Nível de LOG.
- **module_path** Local onde são armazenados os módulos carregados dinamicamente pelo LDAP.
- **moduleload** Nome do módulo usado (forma de armazenamento).
- **sizelimit** Quantidade máxima de linhas que será retornado por consulta.
- **backend** Parâmetros específicos do tipo de base.
- **database** Tipo de base de dados.
- **suffix** O nome da base de dados (domínio).
- **directory** Local de armazenamento dos dados.

Existem outras opções, que serão vistos no momento certo. Por exemplo, as queries de acesso a base de dados, que ficam no final do arquivo. Elas são responsáveis por informar ao servidor como usuários, administradores, etc, terão acesso à base de dados, com que tipo de permissão.

Quando falamos de usuários, no LDAP, devemos lembrar que os usuários no Linux usam quatro arquivos: /etc/passwd, /etc/group, /etc/shadow e /etc/gshadow. Esses arquivos serão substituídos pelos objetos **posixAccount, shadowAccount, posixGroup.**

Todas as classes citadas acima, ficam no arquivo /etc/ldap/schemas/nis.schemas, desta forma, ele deve ter a entrada em /etc/ldap/slapd.conf com um include , que vimos anteriormente.

Configurando o Servidor

Nesta configuração iremos refazer alguns itens, por isso será necessário interromper o processo do LDAP, apagar toda a base, para recriarmos do zero. Desta forma, vocês terão a chance de ver uma segunda forma de configurar, mas de qualquer forma, o arquivo original será mantido, pois apenas dois itens serão alterados.

```
invoke-rc.d slapd stop

rm -rf /var/lib/ldap/*
```

Após isso, devemos editar o arquivo /etc/ldap/slapd.conf, nesse arquivo adicionaremos duas linhas **rootpw** e **rootdn.** Veja abaixo como ficou:

```
rootdn "cn=admin,dc=dominiolinux,dc=net"

rootpw {SSHA}XWbLaW6B5VnQNMg4vVN9ee4lYa4QY+Gp
```

A primeira linha refere-se a quem é o administrador, e a segunda linha qual a senha do administrador. Para gerar essa senha basta executar o comando **slappasswd,** sem nenhum argumento. Será solicitada uma senha, duas vezes. Após isso será exibida a saída com a senha, que deverá ser copiada para o arquivo. No exemplo abaixo digita a senha com um valor debian.

```
slappasswd

New password:

Re-enter new password:

{SSHA}8PvZmHSvN0qRMI3e7aNWdnyEerPKO+qb
```

Agora devemos iniciar o servidor slapd e popular-lo. Para podermos fazer isso, devemos criar uma entrada para o domínio, unidades organizacionais que separarão usuários e grupos, e por fim os usuários e grupos.

Para iniciar vamos criar dois arquivos : **domínio.ldif** e **ou.ldif.** Essa será a base para nosso ldap. Vejamos os arquivos abaixo:

```
# Arquivo dominio.ldif

dn: dc=dominiolinux, dc=net

objectClass: dcObject

objectClass: organization

o: dominiolinux

dc: dominiolinux
```

```
#Arquivo ou.ldif

dn: ou=usuarios,dc=dominiolinux,dc=net

ou: usuarios

objectClass: top

objectClass: organizationalUnit
```

```
dn: ou=grupos,dc=dominiolinux,dc=net

ou: grupos

objectClass: top

objectClass: organizationalUnit
```

Nesses arquivos acima, podemos ver entradas novas como dc (Domain Component), o (organizationName) e os outros já falam por si.

Nesse momento devemos relembrar a ferramenta usada para manipular arquivo LDIF, o **ldapadd.** Vamos adicionar essas entradas, para então começarmos o processo de criação e configuração de usuários no Ldap, autenticando no Linux. E para consulta usamos **ldapsearch.**

```
ldapadd -x -D cn=admin,dc=dominiolinux,dc=net -W -f dominio.ldif

ldapadd -x -D cn=admin,dc=dominiolinux,dc=net -W -f ou.ldif

ldapsearch -x -LLL -b ” -s base ‘(objectclass=*)’ namingContexts

dn:

namingContexts: dc=dominiolinux,dc=net

ldapsearch -x -LLL -b ‘dc=dominiolinux,dc=net’ ‘(objectclass=*)’

dn: dc=dominiolinux,dc=net

objectClass: dcObject

objectClass: organization

o: dominiolinux

dc: dominiolinux

dn: ou=usuarios,dc=dominiolinux,dc=net

ou: usuarios

objectClass: top

objectClass: organizationalUnit

dn: ou=grupos,dc=dominiolinux,dc=net
```

```
ou: grupos

objectClass: top

objectClass: organizationalUnit
```

Nos comandos acima temos duas operações de adição, que é a raiz do sistema e posteriormente as duas Unidades organizacionais. Após os dois comandos temos duas consultas com **ldapsearch,** onde:

- -x Traz no padrão LDIF
- -L remove linhas começando com # (Comentários)
- -b Base Ldap
- '()' Neste parâmetro, é informado o que você quer buscar, no caso nosso todos os objetos.
- -W Solicita senha do administrador LDAP.

Agora já temos a base, vamos dar inicio a criação de usuários e configuração do Linux.

Criando Usuários na Base LDAP

O processo é o mesmo visto anteriormente, ou seja, criaremos um arquivo LDIF e posteriormente iremos inserir via **ldapadd.** A grande diferença são os valores que serão adicionados. Novos campos que são super importante para criação de usuários que irão interagir no Linux.

Então iremos criar um arquivo de conterá tanto o usuário como seu grupo, usando as opções **Posix.** O nome do arquivo será user.ldif.

Caso o sistema já possua contas no Linux, é possível usar o migrationtools para migrar os usuários existentes para uma base Ldap. O processo consiste em formatar uma saída padrão LDIF usando como base o arquivo de usuários, como por exemplo, o /etc/passwd. Depois de instalado, basta executar o comando:

```
cd /usr/share/migrationtools

./migrate_passwd.pl /etc/passwd /root/passwd.ldif
```

Após a execução do comando, o arquivo passwd.ldif terá o mesmo padrão dos arquivos LDIF. Basta editar, alterar o necessário e importar usando o ldapadd.

Agora vamos ao nosso arquivos user.ldif.

```
#arquivo user.ldif

dn: uid=andre,ou=usuarios,dc=dominiolinux,dc=net
```

```
uid: andre

cn: Andre Stato

objectClass: account

objectClass: posixAccount

objectClass: top

objectClass: shadowAccount

shadowLastChange: 15376

shadowMax: 99999

shadowWarning: 7

loginShell: /bin/bash

uidNumber: 10001

gidNumber: 10001

homeDirectory: /home/andre

gecos: Conta Andre,,,

userPassword: {SSHA}XWbLaW6B5VnQNMg4vVN9ee4lYa4QY+Gp

dn: cn=andre,ou=grupos,dc=dominiolinux,dc=net

objectClass: posixGroup

cn: andre

userPassword: {CRYPT}x

gidNumber: 10001

memberUid: andre
```

Neste arquivo temos todas as entradas necessárias para interagir como o Linux. Temos os campos **uid, userPassword (senha), uidNumber, gidNumber, homeDirectory, shadowMax(Tempo máximo sem troca de senha) , shadowWarning (aviso para troca de senha em dias)** e logicamente as classes usadas para isso, **account, posixAccount, shadowAccount.** Os campos falam por si só. Quanto ao grupo, note que é o mesmo gid utilizado anteriormente, e a senha é usado o {CRYPT}x que é o padrão utilizado no /etc/group, onde seria a senha, fica uma letra x. Já no usuário estamos usando md5, que será configurado posteriormente.

O procedimento de adicionar é o mesmo já visto anteriormente.

```
ldapadd -x -D cn=admin,dc=dominiolinux,dc=net -W -f user.ldif
```

Com isso temos nosso cadastro de usuário pronto no Ldap, mas não ainda para o Linux. Próximo passo configurar o cliente ldap e o PAM, responsável por autenticação no Linux.

Configurando Autenticação do Ldap para ser usado pelo PAM

O processo resume-se em instalar o cliente Ldap, configurar o arquivo de resolução de nomes (nsswitch.conf) e configurar o PAM.

Então vamos começar pelas instalações:

```
apt-get install libpam-ldap libnss-ldap nscd
```

O libpam é são módulos ldap para pam, e já os outros dois nscd e o modulo nss, são usado para converter as entradas Ldap para padrão Linux.

Agora temos que configurar os arquivos /etc/libnss-ldap.conf, com a seguinte entrada , baseado no domínio do Post :

```
host 127.0.0.1

base dc=dominiolinux,dc=net

ldap_version 3

rootbinddn cn=admin,dc=dominiolinux,dc=net
```

O arquivo /etc/pam_ldap.conf, é muito similar, contendo uma linha a mais :

```
host 127.0.0.1

base dc=dominiolinux,dc=net

ldap_version 3

rootbinddn cn=admin,dc=dominiolinux,dc=net

pam_password md5
```

Em ambos os arquivos configuramos as entradas para o servidor em host, o domínio, versão ldap, nome do administrador e no caso do pam_ldap, ainda colocamos qual será a criptografia usada, no caso md5.

Outro arquivo importante é o /etc/libnss-ldap.secret, que dera ter a senha do administrador do domínio Ldap. Se você não se sentir a vontade com esse arquivo em

texto puro, altere o local, deixando um link simbólico para o local original, e posteriormente as permissões para 400, sendo o dono o root.

O conteúdo desse arquivo terá a senha somente, sem mais nenhum conteúdo:

```
#libnss-ldap.secret

senha
```

E por fim nesse processo, precisamos dizer para o Linux buscar as informações em uma base Ldap, através do arquivo /etc/nsswitch.conf:

```
passwd: compat ldap

group: compat ldap

shadow: compat ldap
```

Essas três linhas devem ser alteradas, como exibido acima.

Neste momento, já devemos conseguir visualizar a conta que foi criada anteriormente, na base Linux, mas ainda não iremos conseguir autenticar, pois devemos configurar o PAM.

Com o comando abaixo é possível verificar a existência da conta:

```
getent passwd andre

andre:x:10001:10001:Conta Andre,,,:/home/andre:/bin/bash

getent group andre

andre:x:10001:andre
```

Enfim, o ultimo passo é configurar os arquivos do pam. Sugiro fortemente que seja feito backup destes, pois dependendo do erro, você não conseguirá autenticar mais… E com certeza terá necessidade de usar um método de acesso diferenciado (CD-ROM).

Devemos alterar os seguintes arquivos, **/etc/pam.d/common-account, /etc/pam.d/common-auth, /etc/pam.d/common-session , /etc/pam.d/common-passwd.** Abaixo todas as alterações feitas:

```
#common-account

account sufficient pam_ldap.so

account required pam_unix.so
#common-auth
```

```
auth sufficient pam_ldap.so

auth required pam_unix.so nullok_secure
#common-session

session required pam_mkhomedir.so skel=/etc/skel umask=0222

session required pam_unix.so
#common-passwd

password sufficient pam_ldap.so

password required pam_unix.so nullok obscure md5
```

Cada um desses arquivos tem sua importância no processo de logon, Enfim, o ultimo passo é configurar os arquivos do pam. Sugiro fortemente que seja feito backup destes, pois dependendo do erro, você não conseguirá mais logar. E com certeza terá necessidade de usar um método de acesso diferenciado (CD-Rom).

Devemos alterar os seguintes seja na autorização (common-account), esquema de autenticação (common-auth), controle de sessão (common-session, que neste caso criar através do módulo pam_mkhomedir.so o diretório do usuário , caso não esteja criado. Este é opcional, não sendo obrigatório. Mas caso não use, terá que fazer a criação manualmente) e gerenciamento de senhas (common-passwd , tal como tamanho mínimo, horário, etc.)

Agora o processo foi finalizado, o processo de logon deverá funcionar normalmente no Linux, junto com Ldap.

Trabalhoso, mas um ótimo gerenciador e centralizador de autenticação.

Conclusão

Realmente o processo é um pouco complicado, por envolver várias aplicações. Mas obviamente existem outras formas de fazê-lo e ainda ferramentas que podem auxiliar.

Mas tenha certeza que desta maneira funcionará em qualquer plataforma.

Ainda seria necessário alguma alteração no que tange o Ldap, para agilizar o processo de busca, através de criação de indexes, mas isso é assunto para outro Post.

Num próximo Post, veremos gerenciamento de usuários, através das ferramentas disponíveis, tanto as da família ldapclient (ldapadd, ldapsearch, etc.), como ferramentas ldapscripts (vários scripts que nos auxiliam na manutenção da base, e logicamente ferramentas visuais como gOsa e phpLdapAdmin.

Bom, então até a próxima.

André Stato Filho