Montando um Repositorio de Usuarios no LDAP similar a um "Active Directory"

Deives Michellis "thefallen" - William N. Zanata "psych0byte"

---

1. Um pouquinho de Teoria

Vamos dar uma passada (bem por cima) para sabermos mais ou menos o que estaremos implementando aqui.

A autenticacao de usuario num sistema Unix moderno acontece (a grosso modo) quando um programa faz uma chamada a funcao "getpwent" ou similares. Esta funcao consulta a base de usuarios do sistema e retorna uma estrutura contendo os dados referentes ao dito usuario, como username, UID, GID, HomeDirectory, o Shell e, tcharan, o password criptografado do usuario.

Normalmente, basta consultarmos o /etc/passwd e /etc/shadow para obtermos esses dados. Simples nao? Mas, e se eu quiser usar uma outra fonte de dados alem do padrao do sistema? Era uma vez, a muito tempo, uma companhia chamada SUN Microsystems que teve uma ideia muito boa. Por que nao termos uma biblioteca que cuida de obter esses dados? Assim, basta alterarmos a funcao getpwent e similares num unico ponto do sistema, e todas as bibliotecas que usam essa funcao ja poderao estar obtendo dados fornecidos pela nossa biblioteca. Estava nascendo o NameService Switch, uma biblioteca que fornceria servicos pra outras bibliotecas, obtendo dados dos arquivos do sistema, de um diretorio NIS, de um banco de dados Berkeley DB, de qualquer coisa que tivesse um plugin para o NameService Switch. Entre eles, o LDAP. Assim reza o site do NSS_LDAP :)

Assim, podemos usar um repositorio LDAP como "banco de dados" fornecendo informacoes para a biblioteca do NSS, que fornce informacoes para os programas de autenticacao do sistema. "Po, mas quanto intermediario hein?". Eh o preco do sucesso :) De fato, se acessarmos o diretorio LDAP diretamente sem nenhum cache, as consultas ficam um pouco lentas (descobri isso quando fiz "ls -l" ehehe). Pra isso vamos contar com a ajuda do "nscd" - NameService Caching Daemon. Esse cara faz cache das consultas/resultados providos pelo NSS e melhora significativamente a performance do sistema.

Bom, vamos passar entao "ao que interessa" :)

2. Software utilizado

- OpenLDAP - o "proprio" :) - http://www.openldap.org/
- nss_ldap - a biblioteca do NSS que "fala" com o LDAP - http://www.padl.com/OSS/nss_ldap.html
- ladmin - projeto de gerenciador de usuarios - http://www.unitednerds.org/projects/ladmin/

3. Instalacao

Maos a obra!

3.1. OpenLDAP

A instalacao do OpenLDAP eh bem simples. Para nossa instalacao, nao precisaremos de nenhuma opcao especial. Nao precisaremos compilar o OpenLDAP. Se quiser compilar para "ver como eh que eh", sinta-se a vontade, embora isso saia do escopo deste documento. Simplesmente pegue o pacotinho do linuxpackages.net e boa :) Apenas lembrando que o OpenLDAP 2.1.x requere a BerkeleyDB 4.x pra rodar...

Apenas uma dica: voce encontra a Berkeley DB 4 na sessao "extra" do CD/FTP do Slackware.

3.2. nss_ldap

make make install

Vai instalar a biblioteca /lib/libnss_ldap.so.2 e criar os arquivos /etc/nsswitch.ldap e /etc/ldap.conf .

4. Configurando o sistema

4.1. LDAP

Primeiro, vamos definir as conifuguracoes da base do diretorio. Podemos usar o modelo de dcObjects (dc=meudominio, dc=com, dc=br) ou simplesmente o nome da organization (o=dominio).

Normalmente uso o modo de Organization, pois alem de mais simples, eh bem mais facil de digitar na hora de fazer as pesquisas :)

Edite o arquivo /etc/openldap/slapd.conf. Ele precisa conter as seguintes informacoes:

```
/etc/openldap/slapd.conf:
#
# Definicoes de ObjectClass e Attributes
#
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/nis.schema
#
# Local de armazenamento dos dados de PID e afins
#
pidfile         /var/lib/openldap/slapd.pid
```

```
  argsfile        /var/lib/openldap/slapd.args
  #
  # Parametros para fazer o ldappasswd gerar hashes no formato Crypt/MD5
  #
  password-hash {CRYPT}
  password-crypt-salt-format    "$1$%.8s"
  #
  # Permitir que clients mais antigos consigam se conectar ao servidor
  #
  allow bind_v2
  #
  # Modelo de ACLs pra "fechar" legal o acesso ao diretorio
  #
  # nssuser eh o usuario (com poucos privilegios) que o sistema vai usar pra
  # pegar os dados do usuario.
  #
  access to attr=userPassword
          by self +wx
        by dn="cn=nssuser, o=grupogeo" +rsx
          by anonymous auth
  access to *
          by users +xrs
          by anonymous +xrs
  #
  # Tipo de backend que o OpenLDAP vai usar. Por padrao, eh bdb (Berkeley DB) no
OpenLDAP 2.1.x
  #
  base        bdb
  #
  # Nome da organizacao
  #
  suffix          "o=grupogeo"
  #
  # Quem eh o super-usuario do diretorio
  #
  rootdn          "cn=root, o=grupogeo"
  rootpw          MinhaSenhaAqui
  directory       /var/lib/openldap
  #
  # Indices pra agilizar a pesquisa
  #
  index   objectClass     eq
  index   uid             eq
  index   gidNumber       eq
  index   uidNumber       eq
```

Feito isso, precisamos inicializar o diretorio. Inicie o servidor com o comando "slapd" ou "/usr/libexec/slapd" (dependendo de como foi compilado seu LDAP)

Agora precisamos criar a entrada principal do diretorio com o comando "ldapadd" ou "slapadd".

Mais uma dica: Se voce NAO passar a senha correta pro ldapadd, ele vai reclamar. Se a senha estiver correta, ele nao vai aparecer nada, e pode ja comecar a digitar. Finalize com 2 "enters" e um Control D

... #slackware-br was here ...

<PiterPunk> thefallen: toda hora eu... "Putz! Travou! Mas... pôxa, quando eu passo a senha errada ele avisa que deu pau... pq não faz nada quando ponho a senha certa?"

...

```
thefallen@Ragnarok:~ $ ldapadd -D "cn=root, o=grupogeo" -x -W
dn: o=grupogeo
objectClass: top
objectclass: Organization
o: grupogeo
```

Observe que o "o: grupogeo" precisa ser exatamente igual ao "o=grupogeo" do DN (Distinguished Name).

Feito isso, precisamos agora criar as sub-arvores que vao conter os grupos e usuarios.

```
thefallen@Ragnarok:~ $ ldapadd -D "cn=root, o=grupogeo" -x -W
dn: ou=People, o=grupogeo
objectClass: top
objectclass: OrganizationalUnit
ou: People

dn: ou=Users, ou=People, o=grupogeo
objectClass: top
objectclass: OrganizationalUnit
ou: Users

dn: ou=Groups, ou=People, o=grupogeo
objectClass: top
objectclass: OrganizationalUnit
ou: Groups

dn: cn=nssuser,o=grupogeo
objectClass: top
objectClass: person
cn: nssuser
sn: Usuario do NSS_LDAP
```

Vamos agora por uma senha para o usuario nssuser. Este usuario vai ter acesso aos password (criptografados). Capriche!

```
thefallen@Ragnarok:~ $ ldappasswd -D "cn=root, o=grupogeo" -W "cn=nssuser,
o=grupogeo"
```

Pronto. Agora estamos preparados pra deixar o script "ladmin" cuidar do resto :)

Se estiver em duvida quanto ao significado disso que acabou de fazer, veja as definicoes de ObjectClass nos arquivos dentro do /etc/openldap/schemas/ . Sao bastante interessantes.

4.2. nss_ldap

Podemos colocar o diretorio LDAP como um mecanismo de autenticacao primario, unico, ou um "fall back"/suplementar.

Para melhorar a performance do servico, rode o NSCD (Name Service Cache Daemon), que gera caches pros servicos do NS.

Edite o arquivo /etc/nsswitch.conf e maos a obra!

```
/etc/nsswitch.conf:
passwd:         files ldap
shadow:         files ldap
group:          files ldap
```

No caso do slackware, ficaria "passwd: compat ldap". "compat" eh uma macrodefinicao de outros

modulos do nsswitch, incluindo o "files" :)

Este tutorial cobre apenas esseas opcoes, embora seja possivel por praticamente tudo dentro do LDAP.

Precisamos agora informar ao NSS_LDAP qual a senha do usuario "cn=nssuser, o=grupogeo". Esse usuario eh o unico (alem do rootdn) que tem permissao de ler os passwords guardados na base. Edite o arquivo /etc/ldap.secret e coloque a senha la. O arquivo precisa estar com as permissoes 0600.

```
  root@Ragnarok:~# echo MinhaSenhaDoNSSUSER > /etc/ldap.secret ; chmod 0600
/etc/ldap.secret
```

Pra configurar o NSCD, edite o arquivo /etc/nscd.conf e adicione as opcoes a seguir (sao default do NSCD). Mude o debug-level para 0 depois de estabilizar todo o sistema :)

```
  /etc/nscd.conf:
  logfile                 /var/log/nscd.log
  threads                 3
  server-user             nobody
  debug-level             1


  enable-cache            passwd          yes
  positive-time-to-live   passwd          600
  negative-time-to-live   passwd          20
  suggested-size          passwd          211
  check-files             passwd          ye

  enable-cache            group           yes
  positive-time-to-live   group           3600
  negative-time-to-live   group           60
  suggested-size          group           211
  check-files             group           yes
```

Agora eh soh startar o NSCD com o comando "nscd" :-). Certifique-se que ele rodara no proximo boot. Rode o comandinho:

```
  which nscd >> /etc/rc.d/rc.M
```

Vamos agora ao arquivo /etc/ldap.conf, que eh o arquivo que diz ao nss_ldap como lidar com o servidor LDAP. Edite as seguintes opcoes nesse arquivo:

```
  /etc/ldap.conf:
  #
  # O IP (IP, nao hostname; agiliza a pesquisa e, melhor ainda, fica
independente de DNS)
  #
  host 192.168.0.10
  #
  # o "basedn" que voce colocou no /etc/openldap/slapd.conf
  #
  base o=grupogeo
  #
  # O usuario que criamos pra acessar o OpenLDAP
  #
  rootbinddn  cn=nssuser, o=grupogeo
  #
  # Deixe-o procura em toda a sub-arvore do LDAP (base)
  #
  scope sub
```

```
bind_policy hard
```

Pronto! Agora vamos criar nossos usuarios com o "ladmin"

4.3. ladmin

Esse eh um pacote opcional; eh uma ferramentinha que eu montei pra administrar usuarios via comandos do shell. Voce pode perfeitamente usar outras ferramentas, como o YALA ou outros administradores de LDAP. Voce encontra um MONTE no freshmeat.net :)

Vamos primeiro configura-lo. Edite o arquivo /usr/lib/ladmin/config e altere as opcoes necessarias.

```
/usr/lib/ladmin/config:
BASEDN="o=grupogeo"
USERDN="ou=People, o=grupogeo"
GROUPDN="ou=Groups, o=grupogeo"
ROOTDN="cn=root, o=grupogeo"
MINID=20000
HOST=localhost
```

As outras opcoes devem ser deixadas como estao (a menos que voce SAIBA o que esta fazendo...)

Se voce estiver usando OpenLDAP < 2.1.x, de uma olhada no /usr/lib/ladmin/functions . Tem uma linha que precisa ser descomentada pra que tudo "apareca" certinho.

Agora podemos comecar a criar nossos grupos e usuarios.

Rode o comando "ladmin groups add" e crie um grupo no LDAP. Uma "feature" eh que ele soh aceita trabalhar com os grupos que ja estao cadastrados no LDAP (alguem quer converter pra usar o getpwnam e funcoes afins?)

Feito isso, crie usuarios com o comando "ladmin users add".

As opcoes xusers e xgroups servem pra editar os grupos suplementares do usuario.

Sobre o autor

Deives Michellis "thefallen" <dmichellis@yahoo.com | thefallen@unitednerds.org>

- Tecnologo em Processamento de Dados pela FATEC/SP.
- Gerente de Desenvolvimento de Solucoes Linux do Grupo GEO.
- Nerd de carteirinha.

William N. Zanatta aka (psych0byte) - wzanatta@uol.com.br | psych0byte@uol.com.br

- 22 anos, estudante de Engenharia da Computacao pela FASP. Certificado LPIC-2.
- Comecou com computadores ha cerca de 12 anos, soh brincando. Em 1996 conheceu o Linux e o Slackware, unica distribuicao usada desde entao.
- Atua nas areas de desenvolvimento, administracao de sistemas e atualmente tambem como consultor na Veritel Solutions.

Ultima Revisao: Tue Mar 2 20:05:45 2004

Criado com o txt2tags