UNIVERSIDADE FEDERAL DE LAVRAS

INTEGRANDO LDAP COM SAMBA PARA UTILIZAÇÃO COMO SOLUÇÃO DE PDC NA REDE

ADRIANO PINHEIRO MOTA

2008

ADRIANO PINHEIRO MOTA

INTEGRANDO LDAP COM SAMBA PARA UTILIZAÇÃO COMO SOLUÇÃO DE PDC NA REDE

Monografia apresentada ao Departamento de Computação da Universidade Federal de Lavras, como condição prévia para a conclusão do Curso de Pós-Graduação "Lato Sensu" em Administração de Redes Linux.

Orientador

Prof. Samuel Pereira Dias

LAVRAS

MINAS GERAIS – BRASIL

2008

ADRIANO PINHEIRO MOTA

INTEGRANDO LDAP COM SAMBA PARA UTILIZAÇÃO COMO SOLUÇÃO DE PDC NA REDE

Monografia apresentada ao Departamento de Computação da Universidade Federal de Lavras, como condição prévia para a conclusão do Curso de Pós-Graduação "Lato Sensu" em Administração de Redes Linux.

Aprovada em 20 de abril de 2008.

Prof. _____

Prof._____

Prof. Samuel Pereira Dias

(Orientador)

LAVRAS

MINAS GERAIS – BRASIL

Resumo

Esta monografia descreve como configurar e operar o software OpenLDAP que prover serviços de diretório. O LDAP está se tornando uma ferramenta importante na vida de um administrador de redes, então é necessário entende-lo para ser utilizado como solução dentro da rede, no caso especifico esta sendo abortado sua utilização para configuração de um servidor PDC (*Personal Domain Controller*), operando com o Samba integrado com o OpenLDAP, funcionando no sistema operacional Linux. Para centralizar a base de usuários da rede em um único servidor e privilegiar soluções *open-source*, sem custo adicional de conexões e/ou licenças.

SUMÁRIO

LISTAS DE FIGURA

1. INTRODUÇÃO

O propósito principal deste projeto é demonstrar a configuração e operação um Servidor de Diretório LDAP integrado com o serviço Samba, rodando sobre plataforma o sistema operacional Linux. Incluindo detalhes de configuração e execução do LDAP e o processo slapd, e atualizar o processo de replicação slurpd. Também serão consideradas as tarefas de armazenamento, recuperação e atualização das informações do diretório usado pelos clientes do LDAP. Que seja útil para os administradores de redes.

O LDAP (*Lightweight Directory Access Protocol*), pode ter sua utilidade realçada em uma instalação leve, sua funcionalidade é útil para o administrador de redes, que ajuda a evitar que usuários o incomodem perguntando o e-mail de outros usuários.

O LDAP foi projetado para armazenar informações e responder consultas por TCP/IP. Com somente um pouco de informação sobre um usuário registrado (por exemplo, sobrenome ou login) um cliente pode recuperar o número do telefone, endereço de email, etc.

Existe um método para distribuir senhas e outras informações por uma rede é o NIS (*Network Information Service* - Serviço de Informações de Rede). O NIS permite que tenham-se um servidor central para distribuir informações de usuários, incluindo aliases de e-mail e informação de automount. Entretanto, o NIS não é apropriado para administrar grandes objetos binários (BLOBs) como imagens JPEG ou outra coisa que não seja o velho texto plano ASCII. Por outro lado, o LDAP é feito para este tipo de tarefa. Como resultado, o LDAP está sendo usado para substituir serviços NIS no Linux completamente. De fato, o LDAP possui várias vantagens sobre o NIS.

O LDAP possui suporte para listas de controle de acesso (ACLs) para permitir que usuários não-root acrescentem ou modifiquem dados. Suporta criptografia SSL dos clientes, possui chamadas para a codificação das suas próprias aplicações em C ou Perl. A informação do LDAP pode ser utilizada fora do domínio LDAP.

O lançamento do OpenLDAP, oferece aos usuários e administradores de redes, a possibilidade de testar a versatilidade do protocolo LDAP com um código atualizado. OpenLDAP foi codificado para ser escalável, fornecendo suporte para replicação de servidores, e uma escolha de servidores de bancos de dados de backend (como o GDBM).

O projeto do LDAP é hierárquico (como o DNS) ao invés de um banco de dados relacional, o que significa que não é necessário ter nenhum suporte para o mySQL ou o PostgreSQL. Usar uma base de dados relacional diminui um pouco a performance quando se está consultando informações hierarquizadas. A diferença pode não ser muita, mas aumenta conforme a taxa de consultas aumenta.

2. SOFTWARE LIVRE – GNU

As licenças de muitos softwares são desenvolvidas para cercear a liberdade de uso, compartilhamento e mudanças. A GNU (Licença Pública Geral), ao contrário, pretende garantir a liberdade de compartilhar e alterar softwares de livre distribuição, tornando-os de livre distribuição também para quaisquer usuários. A Licença Pública Geral aplica-se à maioria dos softwares da *Free Software Foundation* e a qualquer autor que esteja de acordo com suas normas em utilizá-la.

Quando nos referimos a software de livre distribuição, referimos à liberdade e não ao preço. Licença Pública Geral foi criada para garantir a liberdade de distribuição de cópias de softwares de livre distribuição (e cobrar por isso, caso seja do interesse do distribuidor), o qual recebemos códigos-fontes, que pode ser alterados ou utilizados em parte em novos programas.

Para assegurar os direitos dos desenvolvedores, algumas restrições são feitas, proibindo a todas as pessoas a negação desses direitos ou a solicitação de sua abdicação. Essas restrições aplicam-se ainda a certas responsabilidades sobre a distribuição ou modificação do software.

Finalmente, qualquer programa de livre distribuição é constantemente ameaçado pelas patentes de softwares. Buscando evitar o perigo de que distribuidores desses programas obtenham patentes individuais, tornando-se seus donos efetivos. Para evitar isso, foram feitas declarações expressas de que qualquer solicitação de patente deve ser feita permitindo o uso por qualquer indivíduo, sem a necessidade de licenças de uso.

2.1 SISTEMA OPERACIONAL

O Linux é um sistema operacional criado em 1991 por Linus Torvalds na universidade de Helsinky na Finlândia. É um sistema operacional de código aberto distribuído gratuitamente pela Internet, então praticamente todo mundo pode ter em mãos os códigos-fontes do sistema operacional Linux. Seu código-fonte é liberado como *Free Software* (software livre), o aviso de copyright do kernel (núcleo do sistema) feito por Linus descreve detalhadamente isto e mesmo ele está proibido de fazer a comercialização do sistema. **[Bonan, 2002. pg.03].**

Isso quer dizer que você não precisa pagar nada para usar o Linux, e não é crime fazer cópias para instalar em outros computadores. Ser um sistema de código aberto pode explicar a performance, estabilidade e velocidade em que novos recursos só adicionados ao sistema. **[Bonan, 2002. pg.04].**

Por ser um sistema operacional de código aberto, é possível ver o que código-fonte faz e adaptá-lo às suas necessidades ou de sua empresa. Essa característica é uma segurança a mais para empresas sérias.

Rede TCP/IP mais rápida que no Windows e tem sua pilha constantemente melhorada. O Linux tem suporte nativo a redes TCP/IP

e não depende de uma camada intermediária como o Winsock. Em acessos via modem à Internet, a velocidade de transmissão é dez por cento a vinte por cento maior. **[Bonan, 2002. pg.05].**

O Debian está atento para detalhes que permitem produzir programas de alta qualidade e estabilidade. As instalações podem ser facilmente configuradas para servir múltiplos propósitos , como firewall com poucos pacotes, estações desktop científicas e servidores de rede de alta performance.

Esta distribuição é especialmente popular entre usuários avançados por causa de sua excelência técnica e atenção às necessidades e expectativas da comunidade Linux. O Debian também introduziu muitas características como instalação e remoção fácil de softwares, e também a possibilidade de permitir a atualização do sistema sem requerer a reinstalação**. [Uira, 2004. pg.181].**

Por razões de estabilidade, segurança e custo, o sistema operacional Linux foi escolhido para ser instalado no servidor em questão. Para manter a configuração mínima, sem recursos adicionais e desnecessários, a distribuição Linux escolhida foi o Debian, que oferece, além da simplicidade de configuração e manutenção, a flexibilidade de manter-se apenas o mínimo necessário no sistema. Para o mesmo procedimento poderia ser utilizada outra distribuição do sistema operacional Linux.

3. SAMBA

O SMB – *Server Message Block* – é utilizado pela Microsoft e IBM como padrão de compartilhamento de arquivos. A implementação deste padrão é feita por um pacote de software livre chamado SAMBA.

O Samba é um servidor, é um conjunto de ferramentas que permite que máquinas Linux e Windows se comuniquem e compartilhem arquivos e impressoras. Com ele, é possível construir a topologia de domínios, fazer controle de acesso, compartilhamento, serviço de WINS e outros. **[Uira, 2004. pg.295].**

A primeira versão do Samba, disponibilizada em 1992, foi escrita por Andrew Tridgell, um australiano então estudante de ciências da computação. Como na época a especificação do SMB utilizada pela Microsoft ainda era fechada, Andrew desenvolveu um pequeno programa, batizado de clockspy, para examinar os pacotes de dados enviados por uma máquina Windows e, assim, ir implementando uma a uma as chamadas de sistema utilizadas, um trabalho bastante complexo.

O resultado foi um programa que rodava no Solaris (o sistema Unix desenvolvido pela Sun) e era capaz de responder às chamadas SMB como se fosse um servidor Windows. Este arquivo ainda pode ser encontrado em alguns dos ftps do http: //samba.org, com o nome “Server-0.5”.

O objetivo desta primeira versão era apenas resolver um problema doméstico: interligar um PC rodando o Windows 3.1 à Workstation Sun que ele tinha em casa. Na época isso já era possível utilizando um dos clientes NFS comerciais para DOS, mas Andrew precisava de suporte a NetBIOS para um aplicativo que pretendia

utilizar, o WindX, um servidor X para Windows, que permitia rodar aplicativos via rede a partir do servidor Unix. **[Morimoto, 2006. pg.216].**

Até então o objetivo era apenas fazer o programa funcionar, não criar um sistema de compartilhamento de arquivos. Depois de algum tempo, Andrew recebeu um e-mail contando que o programa também funcionava com o LanManager da Microsoft, permitindo compartilhar arquivos de um servidor Unix com máquinas rodando o DOS. Andrew só acreditou depois de testar, mas ficou tão maravilhado com o que havia conseguido que criou o projeto "NetBios for Unix" e começou a recrutar voluntários através da Usenet. Mais tarde o projeto passou a usar o nome Samba, que foi adotado não em apologia ao carnaval, mas apenas porque é uma das poucas palavras do dicionário do Aspell que possui as letras S, M, B, de "Server Message Blocks".

Em 1994 a Microsoft liberou as especificações do SMB e do NetBios, o que permitiu que o desenvolvimento do Samba desse um grande salto, tanto em recursos quanto em compatibilidade, passando a acompanhar os novos recursos adicionados ao protocolo da Microsoft, que mais tarde novamente deixou de ser aberto.

Hoje, além de ser quase 100% compatível com os recursos de rede do Windows 98, NT e 2000, o Samba é reconhecido por ser mais rápido que o próprio Windows na tarefa se servidor de arquivos.

Um dos pontos fortes do Samba é que o projeto foi todo desenvolvido sem precisar apelar para qualquer violação de patentes. Todas as chamadas (com exceção das que a Microsoft tornou públicas em 1994) foram implementadas monitorando as transmissões de dados através da rede, uma espécie de engenharia reversa que não tem nada de ilegal. E como se descobrisse como funciona um código de encriptação apenas examinando arquivos encriptados por ele. Matemáticos fazem isso a todo instante e muitas vezes são bem pagos para isso. Graças a este "detalhe", o Samba não corre o perigo de sofrer restrições devido a ações judiciais.

De qualquer forma, não existem sinais de que a Microsoft pretenda declarar guerra ao Samba. Pelo contrário, foi a existência do Samba que permitiu que a Microsoft conseguisse colocar PCs rodando o Windows em muitos micros onde só entravam Workstations Unix, já que com o Samba os servidores Unix existentes passaram a ser compatíveis com as máquinas Windows. Ou seja: de certa forma, o Samba foi vantajoso até mesmo para a Microsoft.

O Samba é dividido em dois módulos. O servidor propriamente dito e o cliente, que permite acessar compartilhamentos em outras máquinas (tanto Linux quanto Windows). Os dois são independentes, permitindo que mantenha apenas o cliente instalado num desktop e instale o servidor apenas nas máquinas que realmente forem compartilhar arquivos. Isso permite melhorar a segurança da rede de uma forma geral. **[Morimoto, 2006. pg.217].**

Toda a configuração relacionada com nomes, grupo de trabalho, tipo de servidor, log, compartilhamento de arquivos e impressão do samba está localizada no arquivo /etc/samba/smb.conf. Este arquivo é dividido em seções e parâmetros. As sessões utilizam nomes reservados para configurações específicas. São elas:

[global] Define configurações que afetam o comportamento de todo o servidor Samba, com efeitos em todas os compartilhamentos.
[homes] Especifica opções de acesso aos diretórios HOME dos usuários.
[printers] Define opções gerais para controle das impressoras do sistema. Este compartilhamento mapeia os nomes de todas as impressoras encontradas no /etc/printcap.
[profile] Define um perfil quando o servidor Samba é usado como controlador de domínio.

Outros nomes de sessões podem ser utilizados para definir compartilhamentos de impressoras ou arquivos. **[Uira, 2004 pg.296].**

O Samba possui dois processos que atuam como servidor:

SMBD – Responsável pelos serviços de compartilhamento de arquivos, impressoras e autenticação. **[Uira, 2004 pg.296].**

O processo SMBD verifica a porta 139 (porta usada no samba) e replica-se a cada solicitação do cliente na realização de tarefas de impressão e compartilhamento de arquivos. Se dispõem a cobrir os aspectos de segurança de forma direta. Todo o recurso compartilhado pode estar protegido por uma senha, a qual pode ser implementada de duas distintas: **[Bonan, 2002. pg.373].**

Senhas por compartilhamento: Onde cada item compartilhado como discos, diretórios, impressora etc. tem uma senha própria.
Senhas por usuário: Neste caso todo o usuário deve identificar-se no servidor através de um nome e de uma senha. Após uma validação positiva, o servidor fornecerá os acessos de acordo com as permissões predefinidas para o usuário. **[Bonan, 2002. pg.374].**

NMBD – Responsável pelo serviço de Wins e resolução de nomes em redes SMB. **[Uira, 2004 pg.296].** O processo NMBD recebe todo o tráfego da porta UDP/137 e UDP/138 para os serviços de nomes, registros e *browsing* (navegação). A diferença entre os dois métodos e definida no próprio servidor, não podendo coexistir para um mesmo recurso compartilhado através da rede. **[Bonan, 2002. pg.374].**

Os pacotes do Samba recebem nomes um pouco diferentes nas distribuições derivadas do Debian e no Fedora o outras distribuições derivadas do Red Hat.

Pacote	**Debian**	**Fedora**
Servidor:	samba	samba
Cliente:	smbclient	samba-client
Documentação:	samba-doc	samba-doc
Swat:	swat	samba-swat

3.1 CONTROLADOR DE DOMÍNIO (PDC)

Em uma pequena rede, manter as senhas dos usuários sincronizadas entre as estações Windows e o servidor Samba não chega a ser um grande problema. No entanto, em redes de grande porte, pode se tornar um procedimento trabalhoso, consumindo tempo considerável em ajustes nas configurações.

Para solucionar o problema, existe a opção de usar o servidor Samba como um controlador primário de domínio (PDC), onde ele passa a funcionar como um servidor de autenticação para os clientes Windows e (opcionalmente) armazena os perfis de cada usuário, permitindo que eles tenham acesso a seus arquivos e configurações a partir de qualquer máquina onde façam logon.

Ao cadastrar um novo usuário no servidor Samba, ele automaticamente pode fazer logon em qualquer uma das estações configuradas. Ao remover ou bloquear uma conta de acesso, o usuário é automaticamente bloqueado em todas as estações. Isso elimina o problema de sincronismo entre as senhas no servidor e nas estações e centraliza a administração de usuários e permissões de acesso no servidor, simplificando bastante o trabalho de administração. **[Bonan, 2002. pg.233].**

4. LDAP

LDAP significa *Lightweight Directory Access Protocol*, ou seja, Protocolo Leve de Acesso a Diretórios. Como o nome sugere, é um protocolo leve para acessar serviços de diretório. O LDAP roda em cima do protocolo TCP/IP ou outras conexões de transferência de serviços.

Basicamente LDAP é um protocolo, que trabalha na camada de aplicação da pilha de protocolos TCP/IP, como o SMTP, HTTP, FTP, TELNET e tantos outros.

Só que esses protocolos são bem conhecidos, seus nomes, suas funcionalidades, os processos que os implementam.

O LDAP é uma definição de protocolo para acesso a bancos de dados especializados nos chamados diretórios. É similar ao SQL no sentido que é uma linguagem para interagir com bancos de dados sem especificar um banco de dados particular. De fato, o banco de dados de suporte ao LDAP é quase sempre um sistema RDBMS geral, como o LDBM ou o Oracle.

Um diretório é um conjunto de informações com atributos semelhantes organizados em uma maneira lógica e hierárquica. O exemplo mais comum é a lista telefônica, que consiste em uma série de nomes (ou de uma pessoa ou organização) organizou alfabeticamente, com um endereço e número de telefone agregados.

Um diretório de LDAP freqüentemente reflete várias políticas, limites geográficos, e/ou organizacionais, dependendo do modelo escolhido. De acordo com o desenvolvimento do LDAP atualmente tendem a usar (DNS) nomes para estruturar os níveis mais altos da hierarquia. Dentro do diretório poderia aparecer entrada representando

pessoas, unidades organizacionais, impressoras, documentos, agrupadas pessoas ou qualquer outra coisa que representa uma entrada da árvore (ou entradas múltiplas).

LDAP é um protocolo cliente-servidor leve para acessar serviços de diretório, especificamente baseado em X.500, que prioriza o TCP/IP ou outra conexão similar.

Um diretório é semelhante a um banco de dados, mas tende a conter mais descritivo, informações baseadas em atributo. As informações em um diretório estão geralmente lida muito mais freqüentemente que é escrito. Os diretórios são afinados para dar resposta rápida a pesquisa de volume ou operações de procura alta. Eles podem ter a habilidade de reproduzir informações extensamente a fim de acrescentar disponibilidade e confiabilidade, enquanto reduzindo tempo de resposta.

Existem muitos caminhos diferentes para prover um serviço de diretório. Os métodos distintos permitem tipos diferentes de informações para serem armazenados no diretório, requisitos de lugar diferente em como aquelas informações podem ser referenciadas e atualizadas, como é protegido de acesso sem autorização, etc. Um pouco de serviços de diretório são locais, provendo serviço para um contexto restringido. Outros serviços são globais, provendo serviço para um contexto muito mais abrangente.

4.1 AVALIAÇÃO DO PROTOCOLO

Um diretório é um conjunto de informações com atributos semelhantes organizados em uma maneira lógica e hierárquica.

Um diretório de LDAP freqüentemente reflete várias políticas, limites geográficos, e/ou organizacionais, dependendo do modelo escolhido. Desenvolvimentos do LDAP atualmente tendem a usar (DNS) nomes para estruturar os níveis o mais altos da hierarquia. Mais fundo, dentro do diretório poderia parecer entradas representando pessoas, unidades organizacionais, impressoras, documentos, agrupa das pessoas ou qualquer outra coisa que representa uma entrada de árvore dada.

- Procura por e/ou recupera entradas no diretório;
- Compara, testa se uma chamada e entrada contém um valor de atributo dado;
- Adiciona uma nova entrada;
- Apaga uma entrada;
- Modifica uma entrada;
- Modifica a DN, reposicionar ou mencionar novamente uma entrada;
- Aborta um pedido prévio;
- Operação Estendida, operação genérica usada para definir outras operações;
- Desconectar, fecha a conexão.

Além do servidor pode enviar "notificações não solicitadas", isso são respostas para qualquer solicitação e houver a necessidade de um aviso prévio.

Um método comum alternado para assegurar a comunicação do LDAP é o uso de um túnel de SSL. Isto é denotado em LDAP URLs usando o esquema de URL "ldaps". A porta default para LDAP acima de é 636. O uso de LDAP acima de SSL era comum em Versão de LDAP 2 (LDAPv2) mas nunca estava padronizado em qualquer especificação formal.

LDAP é definido em termos, mensagens de protocolo são codificadas no formato binário onde usa representações textuais para vários ASN.

4.2 COMO TRABALHA O LDAP

O serviço de diretório de LDAP é baseado em um modelo cliente-servidor. Um ou mais servidores de LDAP contêm os dados compondo a árvore de diretório de LDAP ou LDAP backend banco de dados. Um cliente de LDAP conecta a um servidor de LDAP e pergunta. O servidor responde com a resposta, ou com um ponteiro para onde o cliente pode conseguir mais informações (tipicamente, outro servidor de LDAP). Não importa que servidor de LDAP um cliente conecta , terá a mesma visão do diretório, um nome apresentado para uma referências de servidor de LDAP a mesma entrada iria em outro servidor de LDAP. Isto é uma característica importante de um serviço de diretório global, como LDAP.

4.3 O QUE É UM DIRETÓRIO?

Serviço de diretório é um banco de dados otimizado para ler, navegar e procurar. Os diretórios tendem a conter descritivos, atributos, características e um suporte sofisticado para filtros. Os serviços de diretórios geralmente não suportam complicadas transações de registros, grandes volumes de dados, encontrados atualmente em sistema de banco de dados.

As atualizações dos diretórios são simples e rápidas. São feitos para respostas rápidas de um alto volume de operações de buscas. É possível replicar informações largamente para aumentar a confiança e a disponibilidade do recurso.

O conceito de diretório muitas vezes causa confusão. O verbete diretório na literatura especializada tem vários significados, dependendo do contexto. No contexto de sistemas de arquivo ele possui um significado, no contexto de redes e ambientes distribuídos outro e no contexto de banco de dados um terceiro significado.

Esses significados não são excludentes como a maioria de nós poderia supor a princípio. Nisso surge uma dúvida, porque o verbete “diretório” é usado nesses três contextos? Em um nível mais elementar diretório significa “lista”. E lista nada mais é do que depósitos de

informação. A partir dai podemos entender porque diretório é usado nesses contextos.

Vejamos, diretórios em sistemas de arquivo nada mais é do que um arquivo especial que contem as *lista* dos arquivos pertencentes a esse diretório.

No contexto de redes e ambientes distribuídos, diretório é uma lista que contem informações, quase todos serviços de rede, por exemplo, exigem algum tipo de autenticação, obrigando desta forma que os serviços mantenham um diretório de usuários (uma lista de usuários).

Já no contexto com banco de dados é muito natural, uma vez que lista é na verdade um depósitos de informação.

Basicamente o diretório é uma base de dados especializada com o propósito de prover o acesso rápido aos dados de uma maneira padronizada.

4.3.1 INFORMAÇÕES ARMAZENADAS EM UM DIRETÓRIO

O modelo de informações do LDAP é baseado em entradas. Uma *entrada* é uma coleção de atributos que tem forma geral ou exclusiva e nome distinto - (DN - *Distinguished Name*). O DN é usado para se referir a uma entrada que se pode tomar em um único sentido. Cada atributo de entrada tem um *tipo* e um ou mais valores. Os *tipos* são *strings* de memória, como *cn* para nome comum, ou *mail* para endereço de email. A sintaxe de valores depende dos tipos de atributos. Por exemplo, o atributo *cn* pode conter o valor *Juquinha da Silva*. O atributo *mail* pode conter juquinha@mazombo.br.

4.3.2 ORGANIZAÇÃO DAS INFORMAÇÕES?

No LDAP, as entradas dos diretórios são organizadas de forma hierárquica, como uma estrutura de árvore. Esta estrutura possui um limite organizacional.

As entradas são representadas pelos pais que aparecem no topo da árvore. Abaixo aparecem as entradas dos filhos. Abaixo as entradas que representam os setores agregados aos filhos na estrutura hierárquica da arvore.

A árvore pode ser organizada baseada sob domínio de nomes da Internet (DNS). Esta forma de acesso está se tornando muito popular. LDAP permite você controlar os atributos que são obrigatórios e permite a entradas de novos atributos, chamados de *ObjectClass*. O valor dos *ObjectClass* determinam os chamados *SCHEMA RULES* ou *ESBOÇO DAS REGRAS*. Adiante, serão demonstrados mais detalhes sobre *SCHEMA RULES*.

Geralmente os serviços de diretórios não possuem nenhum tipo de mecanismo de proteção dos dados. O LDAP possui mecanismo de autenticação de clientes, privacidade de dados e serviço de integridade.

O LDAP é um serviço de diretórios baseado no modelo Cliente-Servidor. No servidor é executado um processo chamado de SLAPD.

4.3.3 SERVIDORES DE DIRETÓRIO

Um serviço do diretório é uma aplicação que controla os objetos e seus atributos em um diretório. Com o serviço do diretório, os objetos e os atributos podem estar disponíveis aos usuários e a outras aplicações.

Existem muitos servidores de diretórios hoje em dia, por exemplo, o famoso DNS é um serviço de diretório, outro famoso serviço de diretório é o NIS.

Serviço de diretório é a implementação cliente/servidor para o conceito de diretório, sendo que podemos ter diretórios sem o serviço de diretório. É o caso do livro de endereços dos clientes de e-mail. São diretórios locais. Também é o caso da autenticação, no caso dos em sistemas Linux, o famoso /etc/passwd, que nada mais é, que um diretório (deposito de informações) sobre usuários.

Nos dias atuais existe uma grande necessidade de acessar diretórios remotamente. E foram criados inúmeros diretórios e serviços de diretório para os mais diversos fins.

Como o exemplo, uma aplicação pode usar o serviço do diretório recuperar o e-mail de um empregado específico em uma organização.

Tais aplicações de diretórios permitidos usam o serviço do diretório operando de forma eficiente. Os serviços do diretório são baseados geralmente em uma arquitetura do cliente.

Um cliente do e-mail, por o exemplo, é uma aplicação de diretório permitido comum. A função básica de um serviço do diretório é permitir o armazenamento da informação que pode ser recuperado mais tarde.

4.3.4 OPERAÇÃO

O cliente solicita uma mensagem positiva de ID, e a resposta do servidor tem a mesma mensagem de ID. A resposta inclui um código de resultado numérico que indica sucesso, um pouco de condição de erro ou alguns outros casos especiais. Antes da resposta, o servidor pode enviar outras mensagens com outros dados de resultado - por exemplo, cada entrada achada pela operação de procura é retornada em tal mensagem.

Expande a discussão de respostas de indicação para várias operações, especialmente modificação, por exemplo, onde todos modificam devem ser dirigido das réplicas até um diretório de mestre.

4.4 MOTIVOS PARA SE USAR O LDAP

O crescimento gigantesco das redes e usuários dos últimos anos conduziu ao fato que há em sua maioria de redes diferentes, listas

especializadas com a informação redundante parcial, que pode ser usada freqüentemente.

No caso do servidor web apache tem a capacidade de autenticação, portanto necessita de alguma forma de armazenamento para usuário. O apache utiliza uma série de programas (htpasswd, htdigest) e arquivos (htaccess) para armazenar, adicionar e alterar a sua base de dados de usuários. Que na verdade nada mais é que um diretório.

Já o servidor de arquivos Samba usa outro diretório para armazenar dados sobre os usuários. Alem de usar outra série de programas (smbpasswd, smbadduser) para a manutenção dessa base de dados (diretório).

Outros serviços compartilham a mesma base de dados é o caso do servidor de email postfix, do servidor FTP proftpd e do sistema de login do linux.

Alem disso, aplicações desenvolvidas por terceiros também necessitam de algum tido de diretório, seja local ou remoto.

Imaginem que um desenvolvedor crie um controle de estoque, bem os usuários desse sistema, bem como as permissões são mais lidas do que escritas. Essa aplicação seria muito mais aceita e integrada se a autenticação fosse feita através de um serviço de diretório, eliminado o velho problema de um usuário possuir varias contas e várias senhas, e fazer vários logins em vários sistemas.

O LDAP é capaz de unificar diferentes diretórios em um único diretório, seja para servidores, ou para aplicações clientes. Vários pesos pesados da indústria de tecnologia apostam nessa idéia. A Microsoft com seu Active Directory, a Novell com seu NDS, a SUM com o iPlanet. Todas essas empresas apostam no LDAP como mecanismo centralizador de informações. Uma ótima forma de reforçar a necessidade de servidor de diretórios.

4.5 MOTIVOS PARA NÃO UTILIZAR BANCO DE DADOS RELACIONAL

Notamos que a seguinte estrutura é possível e até, a princípio mais fácil de ser implementada. Centralizar as informações que vários serviços necessitam de banco de dados relacional, como por exemplo, o MySQL ou Postgresql.

Mas apesar disso ser uma boa opção temos que levar em consideração que não são só esses serviços que devem ser centralizados. Temos clientes de e-mail, que na maioria das vezes não dá suporte a banco de dados mas dá suporte a LDAP.

Algumas características do LDAP o tornam uma melhor escolha.

Desempenho nas consultas: O LDAP foi desenvolvido com ênfase na leitura, ou seja, os dados serão lidos rapidamente por um número maior de consultas simultâneas.

Interface: Para outras aplicações tais como consulta a um diretório de

e-mails, a interface de comunicação já está embutida em vários aplicativos do mercado.

Padronização: O LDAP vem se tornando o padrão para a disponibilização de informações em forma de diretório.

Peso: Um servidor LDAP (em especial, o OpenLDAP) é leve e não precisa de hardware "anabolizado" para rodar.

Custo: O OpenLDAP é *free* pela GPL e o no máximo, vai lhe custar algumas horas para configurá-lo e mantê-lo.

Em suma: basta você decidir se o LDAP é realmente, a melhor opção. Ele oferece muito, por muito pouco.

4.6 ORIGEM DO LDAP

O LDAP foi originalmente desenvolvido como um cliente para o X.500, o serviço de Diretório OSI. O X.500 define o Protocolo de Acesso a Diretório (DAP) para os clientes usarem quando estiverem em contato com servidores de Diretório.

O DAP era um protocolo difícil de trabalhar e implementar, e protocolos mais fáceis foram desenvolvidos com a maior parte de sua funcionalidade, mas, com muito menos complexidade.

Então o LDAP passou a ser utilizado como serviço autônomo.

O DAP é um protocolo que roda sobre uma camada OSI completa, e precisa de uma quantidade significativa de recursos computacionais para ser executado.

O LDAP roda diretamente sobre o TCP e fornece a maioria das funcionalidades do DAP, a um custo muito menor.

O DAP é um protocolo que roda sobre uma camada OSI completa, e precisa de uma quantidade significativa de recursos computacionais para ser executado.

O LDAP roda diretamente sobre o TCP e fornece a maioria das funcionalidades do DAP, a um custo muito menor.

As companhias de telecomunicação introduziram o conceito de serviços de diretório e como seus compreensivos requisitos de diretório estavam bem desenvolvidos depois de alguns anos de produção e administrando listas telefônicas.

Os serviços de diretório de X.500 estavam tradicionalmente acessados via o X.500 (DAP), que exigiu o (OSI) pilha de protocolo. LDAP era originalmente com intenção de ser um "protocolo alternativo de peso leve” para acessar serviços de diretório de X.500 pelo mais simples (e agora difundido) pilha de protocolo. Este modelo de acesso de diretório era obtido emprestado dos protocolos.

Servidores de diretório de *standalone* LDAP logo seguido, como fizeram servidores de diretório suportando ambos os DAP e LDAP. O posterior ficou popular em empreendimentos, como LDAP removeu qualquer precisar desdobrar uma rede de OSI. Hoje, protocolos de diretório de X.500 inclusive DAP também podem ser diretamente usados acima de TCP/IP.

O protocolo era originalmente criado por Wengyik Yeong. No início de fases de engenharia de LDAP, era conhecido como *Directory Light Browse Protocolo*, ou *LDBP*. Era mencionado novamente como o escopo do protocolo era expandido para não incluir só diretório browse e funções buscadores, mas também diretório atualiza funções.

LDAP influenciou protocolos de Internet subseqüente, inclusive versões mais velhas de X.500, (XED), (DSML), (SPML), e o (SLP).

4.7 LDIF (Lightweight Data Interchange Format)

Este é um formato texto de intercâmbio de informações para o LDAP. Tal formato foi definido para que possamos, humanamente entender as entradas do diretório quando de sua geração ou de sua exportação para um arquivo texto.

Uma entrada escrita em tal formato:

dn: cn=Maria José da Silva, o=mazombo, c=BR
objectclass: person
cn: Maria José da Silva
cn: Maria
sn: Silva

Essas entradas são escritas em um editor de texto comum como o vi (UNIX) ou edit (Windows) e gravado com a extensão ldif

Com o comando de adição ldapadd incluímos essa entrada no banco de dados do openldap

ldapadd -x -v -W -D ”cn=manager,o=mazombo,c=br” -f entrada.ldif'

Embora essa entrada seja somente de um objeto podemos referenciar vários objetos ao mesmo tempo.

4.8 DIRETÓRIO REPLICADO

A grande maioria dos usuários de um sistema distribuído já teve a oportunidade de experimentar problemas em relação ao tempo de acesso as informações.

Tais problemas são decorrentes de diversos fatores como a limitação da largura de banda, grande número de usuários acessando o sistema simultaneamente e o aumento do tamanho das informações transferidas, como por exemplo, vídeos, imagens e sons.

Para reduzir o tempo de acesso a tais informações uma possível solução é a replicação. Replicação de dados é a criação e a manutenção de cópias de uma base de dados ou sistema de arquivos. Estas cópias são mantidas em servidores independentes para aumentar a confiabilidade e garantir acesso local.

Com somente um servidor LDA, temos sobrecarga, uma vez que todos utilizam o mesmo servidor, e insegurança, já que se, por algum

motivo, o servidor parar todos ficam impossibilitados de acessas as informações.

4.8.1 DIRETÓRIO DISTRIBUIDO

Um serviço de diretórios pode estar distribuído em diferentes endereços físicos. Cada servidor é responsável apenas por uma parte do diretório (uma sub-árvore ou partição). Desta forma, podemos melhorar o desempenho e a escalabilidade do serviço.

4.9 MODELOS LDAP

O LDAP define quatro modelos básicos que descrevem por completo a sua operação, que informações podem ser armazenadas em diretórios LDAP e o que pode ser feito com essas informações.

Modelo de Informação: Define o tipo de informação que pode ser armazenada em um diretório LDAP.

Modelo de Nomes: Define como a informação no diretório LDAP pode ser organizada e referenciada.

Modelo Funcional: Define o que pode ser feito com a informação no diretório LDAP e como ela pode ser acessada e alterada.

Modelo de Segurança: Define como a informação no diretório LDAP pode ser protegida de acessos ou modificações não autorizadas.

4.9.1 MODELO DE INFORMAÇÃO

Define o tipo de informação que pode ser armazenada em um diretório LDAP.

A unidade básica da informação armazenada no diretório é chamada uma entrada. Estas representam objetos de interesse no mundo real tal como, usuários, organizações, e assim por diante.

Entradas são coleções dos atributos e dos seus valores. Cada atributo tem um tipo e um ou mais valores.

Um objeto, para ser inserido no contexto de um diretório, precisa ter sua forma definida, ou seja, uma pessoa (objeto da classe person) requer certos atributos e permite outros.

objectclass person
requeridos
cn,
sn,
objectClass
permitidos
seeAlso,

description,
telephoneNumber,
userPassword

4.9.2 MODELO DE NOMES

Define como a informação no diretório LDAP pode ser organizada e referenciada.

O modelo de nomes pressupõe um diretório organizado em árvore. As entradas do diretório são os nós da arvore. Cada entrada tem um *relative distinguished name* (RDN), que é relativo ao seu nó pái, e um *distinguished name* (DN), que especifica um, caminho da raiz até a entrada.

dn: ps=maria, ou=secretaria,o=mazombo, c=br

No LDAP, as entradas de diretório são organizadas em uma estrutura hierárquica como uma árvore que reflete limites políticos, geográficos e/ou organizacional.

4.9.3 MODELO FUNCIONAL

Define o que pode ser feito com a informação no diretório LDAP e como ela pode ser acessada e alterada.

Funcionalmente o LDAP define nove operações, divididas em três categorias:

Interrogação (consultas com filtros)

- Buscar;
- Comparar.

Essas operações são usadas para consultar o diretório e recuperar
as informações nele armazenadas.

Atualização

- Adicionar;
- Apagar;
- Modificar;
- Renomear (modificar RDN);
- Essas operações são usadas para alterar as informações no diretório.

Autenticação e controle

- Associar (bind);
- Dissociar (unbind);
- Abandonar.

As operações bind e unbind são de gerenciamento de sessão. A operação abandonar permite cancelar uma operação em processo.

4.9.4 MODELO DE SEGURANÇA

Define como a informação no diretório LDAP pode ser protegida de acessos ou modificações não autorizadas.

Existem três aspectos básicos na proteção de informação em um diretório:
acesso, autenticação e autorização (AAA, ou Triplo-A).

ACESSO é a habilidade de conectar-se a um serviço e pode ser restringida baseada em detalhes como hora do dia ou endereço IP. Para acesso seguro, o LDAP suporta o *Transport Layer Security* (TLS), que criptografa toda a comunicação entre cliente e servidor. Desta forma garante a confiabilidade das informações que trafegam na rede.

AUTENTICAÇÃO é a habilidade de provar ao serviço que um cliente é um usuário válido. Para autenticação, o LDAP suporta a *Simple Authentication and Security Layer* (SASL), que permite que o cliente e servidor negociem um método de autenticação (seguro).

AUTORIZAÇÃO é o serviço fornecendo ou negando direitos específicos ou funcionalidades ao cliente. A autorização é controlada pelas ACLs.

O LDAP fornece a habilidade de controlar todos os três aspectos da AAA através de *Access Control Lists* (ACLs) (Listas de Controle de Acesso).

As ACLs podem ser usadas para autorizar o acesso baseado em muitos fatores diferentes. Elas podem ser usadas para forçar tipos específicos de autenticação e, uma vez que o cliente esteja plenamente autenticado como usuário válido, as ACLs são usadas para autorizar o usuário.

O cliente, quando chama a operação bind, fornece sua identificação (um *distinguished name*) e credenciais de autenticação, senhas, chaves privadas, etc. Uma lista de controle de acesso é usada para determinar quais entradas dos diretórios o cliente pode ver que alterações ele tem permissão para fazer.

Há a possibilidade do usuário não se identificar, ou seja, acessar o diretório como anônimo. Nesse caso as regras de controle de acesso também determinarão o que o usuário poderá acessar no diretório.

4.10 SLAPD

SLAPD é um servidor de diretórios LDAP que roda em vários tipos de plataformas. Utilizado para criar os serviços de diretório. Em cada diretório pode conter muitas informações interessantes, que você mesmo pode administrar. O SLAPD pode servir para um serviço de diretório global ou para apenas você utilizar. Algumas das características importantes do SLAPD são:

SLAPDv3 - O Slapd versão 3 possui suporte a IPv4 e IPv6;

Simples autenticação segura - O Slapd suporta uma forte segurança através do uso de SASL, o qual suporta mecanismos como MD5, EXTERNAL e GSSAPI, suporte a SSL.

Controle de acesso - slapd provê um acesso rico e poderoso controlarem instalação, permitindo a você controlar acesso às informações em seu banco de dados(s). Pode-se controlar acesso a entradas baseadas em informações de autorização de LDAP, endereço de IP, nome de domínio e outros critérios. Slapd suporta ambos, estáticos *e* acesso dinâmico controlarem informações.

Controle de topologia - slapd pode ser configurado para restringir acesso na camada de soquete baseado em informações de topologia de rede. Esta característica utiliza *envolturas de TCP*.

Internacionalização - slapd suporta unicode e etiquetas de idioma.

Diversificação de banco de dados - O Slapd vem com diversas opções de banco de dados. Incluindo BDB, LDBM, via SHELL ou PASSWD. Permite vários bancos de dados no mesmo servidor.

Api de Módulos genéricos - Se for necessário mais customização, *slapd* deixa escrever seus próprios módulos facilmente. O *slapd* consiste em duas partes distintas: uma frente termina aquela comunicação de protocolo de handles com clientes de LDAP; e módulos que tarefas de handle específico como operações de banco de dados. Porque estes dois pedaços comunicam via bem definida C API, pode escrever seus próprios módulos personalizado, que estendem *slapd* em modos numerosos. Também, vários módulos de banco de dados programáveis são providos. Estes permitem expor origens de dados externas para slapd usando idiomas de programação popular (Perl, Shell, SQL e TCL). Possui *threads* para alto desempenho.

Replicação - o slapd permite ser copiado para outros servidores, podendo ter servidor master e slave; É configurado através de um único arquivo de configuração. Este único-*master*/*multiple*-escravo esquema de replicação é vital em ambientes de volume alto onde um único *slapd* só não provê a disponibilidade ou confiabilidade necessária. *Slapd* suporta dois métodos de replicação: *LDAP Sync* e *slurpd*.

Transporte Segurança de Camada: slapd suporta autenticação e segurança de dados baseados em certificado (integridade e confidência) serviços pelo uso de TLS (ou SSL). Slapd é implementação de TLS utiliza o software OpenSSL.

Escolha de banco de dados backends: slapd vem com uma variedade de banco de dados diferente backends que pode se escolhida. Eles incluem BDB, um alto desempenho transacional, banco de dados backend; HDB, um alto desempenho hierárquico transacional backend; LDBM, um DBM de peso leve baseado backend; *Shell*, um backend interface para escrituras de Shell arbitrária; e PASSWD, um simples backend interface para o arquivo *passwd*. O BDB e HDB *backends* utilizam *Sleepycat* DB de Berkely. LDBM utiliza um ou outro DB de Berkeley ou GDBM.

Instâncias de banco de dados múltiplo: slapd pode ser configurado para servir para bancos de dados múltiplos ao mesmo tempo. Isto significa que um único *slapd* servidor pode responder para pedidos para muitas porções logicamente diferentes da árvore de LDAP, usando o banco de dados mesmo ou diferente backends.

4.11 SLURPD

Slurpd é um processo para ajudar o slapd a replicar seus serviços. É responsável pela distribuição das modificações do servidor master para os outros servidores. O slapd e o slurpd se comunicam através de um arquivo comum de texto.

O arquivo slapd.conf está dividido em três partes de configuração: global, backend e banco de dados. Os comentários podem ser definidos por '**#**' e se a linha começa com um espaço em branco. O formato geral do arquivo slapd.conf é definido assim:

 # Diretivas globais de configuração
global

 # Definição de backend
backend

 # Primeiro banco de dados e configuração das diretivas
database

 # Segundo banco de dados e diretivas
database

A configuração das diretivas precisa ter argumentos, por isto que elas são separadas por espaço. Se um argumento possui um espaço em branco, o argumento precisa ser ter aspas duplas. "Desta forma". A distribuição contém um exemplo de configuração do arquivo slapd.conf, que será instalado no diretório /etc/ldap. O número de arquivos

contendo as definições do *SCHEMA* (atributos e os *ObjectClass*) também estará disponível no diretório /etc/ldap/schema.

4.12 DIRETIVAS DO ARQUIVO DE CONFIGURAÇÃO

As diretivas geralmente utilizadas no arquivo slapd.conf e dividido em diretivas globais, diretivas de backend e diretivas de banco de dados.

Cache de procuração: *slapd* pode ser configurado como um *caching* LDAP procuração do serviço.

Configuração: slapd é altamente configurável por um arquivo de configuração única que permite mudar quase tudo. As opções de configuração têm *defaults* razoáveis, fazendo seu trabalho muito mais fácil.

4.12.1 DIRETIVAS GLOBAIS

access to by +

Esta diretiva permite o acesso (*accesslevel*) para um grupo de entradas e/ou atributos (especificado pelo) por um ou mais solicitadores. Se nenhuma diretiva de access é especificada, o padrão de acesso é * by * read, permitindo que tanto usuários para autenticar ou anônimos tem acesso de leitura.

attributetype

Especifica o tipo de atributo.

idletimeout

Especifica o número de segundos para esperar antes de fechar a conexão com o cliente. Se for 0, desabilita esta opção.

include

Esta diretiva especifica que o slapd precisa adicionar informações adicionais de um outro arquivo. Este arquivo precisa estar no formato de configuração do slapd. Geralmente é utilizado para incluir as especificações dos Shemas.

loglevel

Esta diretiva especifica o nível de log para apresentar no syslog. O padrão é 'loglevel 256', onde apresenta o status de conexões, operações e resultados. O parâmetro '-1' no loglevel habilita todos os logs do OpenLDAP no syslog.

objectclass

Esta diretiva define um objectclass.

referral URL

Esta diretiva especifica a orientação para passar quando o slapd não conseguir achar um banco de dados.

sizelimit

Esta diretiva especifica o número máximo de entradas para retornar em uma operação de procura.

timelimit número

Esta diretiva especifica o número máximo de segundos que o slapd leva para responder uma requisição de procura.

4.12.2 DIRETIVAS DE BANCOS DE DADOS

As diretivas são aplicadas apenas aos banco de dados compatíveis.

database

Esta diretiva marca o inicio do banco de dados.

readonly { on | off }

Esta diretiva põe o banco de dados em modo "read-only", apenas leitura. Qualquer tentativa de modificar o banco de dados retornará erro.

replica

Diretiva responsável para replicar os dados para outro banco de dados. Esta diretiva possui vários parâmetros.

replogfile

Especifica o nome de log para réplica.

rootdn DN

Coloca-se o nome e o dominio do administrador do diretório Ldap.

rootdn "cn=ldap,dc=mazombo,dc=br"

rootpw

Diretiva para especificar a senha do DN para o rootdn. A senha pode também ser criptografada, utilizando o comando slappasswd -s senha.

rootpw {SSHA}ZKKuqbEKJfKSXhUbHG3fG8MDn9j1v4QN
ou
rootpw secret

suffix

Diretiva que especifica o sufixo DN de pesquisas que serão transmitidas para este banco de dados.

suffix "dc=mazombo,dc=br"

4.13 LDAP BACKENDS, OBJETOS E ATRIBUTOS.

O processo do servidor de LDAP é chamado slapd. Slapd suporta uma variedade de banco de dados diferente *backends* que você pode usar.

Eles incluem a escolha primária BDB, um alto desempenho do banco de dados backend; LDBM, um DBM leve baseado em backend; shell, uma interface backend para escrituras do Shell arbitrária e PASSWD, um simples interface backend para o arquivo passwd.

Para importar e exportar informações de diretório entre servidores de diretório baseado em LDAP, ou descrever um conjunto de mudanças que serão aplicadas a um diretório, o formato do arquivo conhecido como LDIF, para Formato de Intercâmbio de Dados de LDAP, é tipicamente usado. Informações de arquivo LDIF são orientadas a hierarquias de objeto de entradas. O pacote de software de LDAP consegue-se um utilitário para converter arquivos de LDIF para o formato de BDB.

Um arquivo de LDIF comum parece com isto:

dn: o=mazombo, c=BR
o: mazombo
objectclass: organization
dn: cn=Maria Jose da Silva, o=mazombo, c=BR
cn: Maria Jose da Silva
sn: Maria
mail: maria@mazombo.com
objectclass: person

Cada entrada está exclusivamente identificada por um nome distinto, ou DN. O DN consiste no nome da entrada mais um caminho de nomes localizando a entrada de volta para o topo da hierarquia de diretório.

No LDAP, uma classe de objeto define a coleção de atributos que podem ser usados para definir uma entrada. O padrão de LDAP provê estes tipos básicos de classes de objeto:

- Agrupa no diretório, listas de objetos individuais ou agrupa os objetos.
- Posições, como o nome e descrição.
- Organizações no diretório.
- Pessoas no diretório.

Uma entrada pode pertencer a mais de uma classe de objeto. A entrada para uma pessoa é definida pela classe de objeto de pessoa, mas pode também ser definida por atributos no inetOrgPerson,

groupOfNames, e organização objectclasses. A estrutura de classe de objeto do servidor determina a lista total exigida e permite atributos para uma entrada particular.

Os dados de diretório é representado como atributo. Qualquer pedaço específico de informações é associado com um atributo descritivo.

O commonName, ou cn, atributo é usado para armazenar nome da pessoa . Uma pessoa chamada Maria Silva pode ser representada no diretório como:

cn: Maria Silva

Cada pessoa que entra no diretório é definido pela coleção de atributos na classe *de objeto de pessoa*. Outro atributo usado para definir esta entrada pode ser:

givenname: Maria
surname: Silva
mail: maria.silva@mazombo.com**.br**

O atributo exigido inclui os atributos que devem estar presentes em entradas usando a classe de objeto. Todas as entradas exigem o objectClass atributo, que lista os classes de objeto para que uma entrada pertence.

Cada atributo tem uma definição de sintaxe correspondente. A definição de sintaxe descreve o tipo de informações providas pelo atributo.

Normalmente objectclass e definições de atributo residem em arquivos de esquema, no esquema de subdiretório debaixo do home da instalação de OpenLDAP.

4.14 STARTLS

A operação de StartTLS estabelece (o descendente de SSL) na conexão. Isso pode prover confidência de dados e/ou proteção de integridade dos dados. Durante a negociação de TLS o servidor envia seu certificado para provar sua identidade. O cliente pode também enviar um certificado para provar sua identidade. Depois de fazer isso, o cliente pode então usar /external para ter esta identidade usada em determinar a identidade usada em fazer decisões de autorização de LDAP.

Os servidores também freqüentemente suportar o não normal "LDAPS" ("LDAP SEGURO", comumente conhecido como "LDAP acima de SSL") protocolo em uma porta separada, à revelia 636. LDAPS difere de LDAP em dois modos:

1) conecta, o cliente e servidor estabelecem TLS antes de quaisquer mensagens de LDAP serem transferidas (sem uma operação de Começo TLS).
2) a conexão de LDAPS deve ser fechada em encerramentos de TLS.

LDAPS foi principalmente usado com LDAPv2, porque a operação de StartTLS ainda não tinha sido definida.[wikipedia].

4.15 BIND (autentique)

Vincula operação que autentica o cliente para o servidor. Simples vinculação pode enviar o usuário DN e senha, então a conexão deve ser protegida usando (TLS). O servidor tipicamente checa a senha contra o userPassword atributo na chamado entrada.

Vinculação anônima (com DN vazio e senha) reajustar a conexão para estado anônimo. (Autenticação e Camada de Segurança simples) O bind provê serviços de autenticação por uma grande variedade de mecanismos ou o certificado de cliente enviou com TLS.

O bind também fixa a versão de protocolo de LDAP. Normalmente clientes deveram usar LDAPv3, que é o default no protocolo.

O bind teve que ser a primeira operação em uma sessão em LDAPv2, mas não é exigido em LDAPv3 (a versão de corrente LDAP).

4.16 PROCURA E COMPARA

A operação de procura é usada para ambas as procura para lê entradas. Seus parâmetros são:

baseObject - o DN (Nome Distinto) da entrada em que começar a procura.

escopo - baseObject (procure apenas da chamado entrada, tipicamente usada para ler uma entrada).

singleLevel - (entradas imediatamente abaixo da base DN).

wholeSubtree - (a subárvore inteira começando na base DN).

filtro - como examinar cada entrada no escopo. Por exemplo (&(objectClass=pessoa)(|(givenName=Maria)(remeta=Maria*))) - procura por pessoas que ou deram nome Maria ou um endereço de e-mail que começa com Maria.

derefAliases - e como seguir entradas de nome alternativo (entradas que se referem a outras entradas).

atributos - atributos para retornar para entradas de resultado.

sizeLimit, timeLimit – Máximo de número de entradas, e máximo tempo de procura.

typesOnly - tipos de atributo de retorno somente, não valores de atributo.

O servidor retorna as entradas de comparação e talvez referências de continuação (em qualquer pedido), seguido pelo final resulta com o código de resultado.

Comparar operação toma um DN, um nome de atributo e um valor de atributo, e checa se a chamado entrada contém aquele atributo com que estime.

4.17 ATUALIZAÇÃO DAS OPERAÇÕES

As tarefas de adicionar, apagar e modificar DN exige o DN da entrada que é para ser mudada.

Modifica-se uma lista de alterando cada um de seus atributos específicos.

Apaga-se o atributo ou alguns valores, adicione novos valores, ou substituir os valores de corrente com a nova.

Adicionar operações também podem ter atributos e valores adicionais ou alterados.

Modificação do DN (move/renomea entrada) tome o novo RDN (Nome Distinto Relativo), opcionalmente o novo pai é DN, e uma *flag* que diz se apagar o valor(s) na entrada que combina o RDN velho. O servidor pode suportar mencionando novamente de subárvores de diretório inteiras.

Uma operação de atualização é atômica: Outras operações serão a nova entrada ou a velha. Por outro lado, LDAP não define transações de operações múltiplas: Se ler uma entrada e então modificar, outro cliente pode ter atualizado a entrada no tempo médio. Os servidores podem implementar extensões que suportam isto.

4.18 OPERAÇÃO EXTENDIDA

A Operação Estendida é uma operação de LDAP genérico que pode ser usado para definir novas operações. Os exemplos incluem o cancelar, modificar senha e começar operações de TLS.

4.18.1 ABANDONO

A operação abandonar pedidos que o servidor aborta uma operação chamada por uma mensagem ID. O servidor não precisa honrar o pedido. Infelizmente, nem Abandone nem uma operação com sucesso abandonado envia uma resposta. Um semelhante cancelar operação estendida, então foi definido que envia respostas, mas nem todas as implementações suportam isto.

4.18.2 UNBIND

A operação unbind abandona quaisquer operações excedentes e fecha a conexão. Não tem nenhuma resposta. O nome é de origem histórica: Não é o oposto da vinculação da operação. Os clientes podem abortar uma sessão simplesmente fechando a conexão, mas eles deveram usar desconectar. Caso contrário o servidor pode dizer à diferença entre uma conexão de rede falha (ou um ataque de mutilação) e um cliente descortês.

4.18.3 SCHEMA

O schema define os tipos de atributo que as entradas de diretório podem conter. Uma definição de atributo inclui uma sintaxe, e valores mais não binário em LDAPv3 usam sintaxe de string. Um "atributo de correio poderia conter o valor "jose@mazombo.br". Um "atributo de membro contém DNS de outras entradas de diretório. As definições de atributo também especificam se o atributo é único-estimado ou estimado múltiplo, como procura ou compara o atributo.

O schema define *classes de objeto*. Cada entrada deve ter um objectclass atributo, contendo classes definidas no schema. A definição de esquema das classes de uma entrada define o que um tanto quanto o objeto e a entrada podem representar, como uma pessoa, organização ou domínio. As definições de classe de objeto também listam atributos que a entrada deve conter. Uma entrada representando uma pessoa poderia pertencer ao topo de classes" e "Sociedade de pessoa no "classe de pessoa exigiria a entrada para conter o "sn" e "cn" atributos, e permitam a entrada também para conter "userPassword", "telephoneNumber", e outros atributos. Desde entradas podem pertencer a classes múltiplos, cada entrada tem um complexo de atributo opcional e obrigatório fixa formado da união dos classes do objeto.

O schema também inclui várias outras informações que controla as entradas de diretório.

A maioria de elementos do schema tem um nome e um globalmente diferente.

Os servidores de diretório podem publicar o schema do diretório controlando uma entrada em uma base DN dado pelo atributo *subschemaSubentry* operacional da entrada. (Um *atributo operacional* descreve operação do diretório em lugar de informações de usuário e está só retornada de uma procura quando for explicitamente solicitado).

Os administradores do servidor podem definir seus próprios schemas além dos padrões. O esquema define os *tipos de atributo* que entradas de diretório podem conter. Um atributo de membro contém DNS de outras entradas de diretório.

4.18.4 VARIAÇÕES

A operação de servidor é remanescente para o implementador ou administrador decidir. Conseqüentemente, servidores podem ser instalados para suportar uma larga variedade de argumentos.

Memória de dados no servidor não é especificada, o servidor pode usar arquivos simples, bancos de dados, ou só são um portal para algum outro servidor. O controle de acesso não é padronizado. As senhas dos usuários podem ser armazenadas em suas entradas ou em outro lugar. O servidor pode recusar apresentar operações quando desejar, e impor vários limites.

A maioria de partes de LDAP são extensível. Pode se definir novas operações. *Os controles* podem modificar pedidos e respostas, solicitar classificar resultados de procura. Novos escopos de procura e vinculação de métodos podem ser definidos. Os atributos podem ter *opções* que podem modificar sua semântica.

4.19 OUTROS MODELOS DE DADOS

Como LDAP tem como oferecer um protocolo de acesso para outros serviços. A implementação então reforma os dados para imitar o modelo de LDAP/X.500, mas como próximo modelo varia. Existe software para acessar bancos de dados por LDAP, embora LDAP não é propriamente para este fim. Semelhantemente, dados que estavam previamente seguros em outros tipos de alojamento de dados, estão às vezes reposicionados para diretórios de LDAP. Como usuário unix e agrupar informações podem ser armazenadas em LDAP e acessado via módulos. LDAP é freqüentemente usado por outros serviços para autenticação.

4.19.1 APLICATIVOS

Deste modo, se escolhe alguns protocolos gerais como LDAP e para vários serviços, podem se enfocar nestes poucos protocolos em vez de ter que manter e melhorar muitos protocolos especializados.

Aplicativos comuns de LDAP são para computadores, usuários e grupos. Muitos clientes de e-mail suportar pesquisas de LDAP.

4.19.2 NOMEANDO ESTRUTURA

Um servidor de LDAP pode retornar indicações para outros servidores para solicitar o servidor que propriamente checa a estrutura, nomear entradas para o LDAP, é preciso poder achar um servidor segurando um dado DN. Desde que tal estrutura já exista no (DNS), nível superior dos servidores.

Se uma organização tem nome de domínio mazombo.br, seu nível superior entrada de LDAP então tipicamente terá o dc de DN=mazombo,dc=br (onde dc significa componente de domínio). Se o

servidor ldap também é chamado ldap.mazombo.br, o nível de topo da organização que URL de LDAP se torna ldap://ldap.mazombo.br / dc=mazombo,dc=br.

Abaixo do nível superior, a entrada tipicamente nomeia refletirá a estrutura ou necessidades internas da organização em lugar do DNS nomeado.

4.19.3 TERMINOLOGIA

A terminologia de LDAP que se pode encontrar é bastante confusa. Algumas dessas confusões é devido a enganos, outro exemplo é devido a suas origens históricas, outros surjam quando usados com serviços que usar terminologia diferente.

O "LDAP" é às vezes usado para se referir ao protocolo, outros tempos para o protocolo e os dados. Um "DIRETÓRIO de LDAP" pode ser os dados ou também o ponto de acesso. Um "atributo" pode ser o tipo de atributo, ou o conteúdo de um atributo em um diretório, ou uma descrição de atributo (um tipo de atributo com *opções*). Umas "anônimas" e umas "não autenticadas" vinculações são diferentes. Vincular métodos que ambos os estado de autenticação de produto anônimo, então estão sendo usadas para ambas as variantes. O "uid" atributo devia segurar nomes dos usuários em lugar de usuário numérico IDs.

4.19.4 ACL's

Deve-se restringir o acesso a escrita das entradas à conta Manager, exceto para a sub-árvore do usuário. Esta sub-árvore também deverá conter direitos de escrita para o administrador. O administrador está habilitado a criar novos usuários, modificar os atributos dos usuários existentes e deletar usuários. A primeira ACL assegura que os usuários possam modificar suas senhas com seus próprios privilégios. Para estar de acordo com a empresa, simplesmente substituir “mazombo” e “br” pelos nome de domínio da empresa no arquivo slapd.conf. Reinicia o servidor OpenLDAP após o término da edição do arquivo slapd.conf.

```
access to attr="userPassword"
 by self write
 by dn="cn=Manager,dc=mazombo,dc=br" write
 by dn="cn=admin,dc=mazombo,dc=br" write
 by anonymous auth
 by * none
access to *
 by dn="cn=Manager,dc=mazombo,dc=br" write
 by dn="cn=admin,dc=mazombo,dc=br" write
 by * read
```

Descrição dos Privilégios:
none=0 ***sem acesso***
auth=x ***necessários bind***
compare=cx ***necessário para comparar***
search=scx ***necessários para aplicar filtros de procura***
read=rscx ***necessário para ler resultados das buscas***
write=wrscx ***necessários para modificar/renomear***

4.19.5 THREADS

Os threads tem suporte garantido para ser parte da base do sistema Linux. O OpenLDAP é projetado para aproveitar-se das threads. OpenLDAP suporta POSIX pthreads, C Threads, e várias outras variedades. O ato de configurar, requisitará a possibilidade de poder achar um subsistema de thread apropriado.

4.19.6 TCP WRAPPERS

Slapd suporta TCP Wrappers (IP nivela o acesso ao controle de filtros). Use TCP Wrappers ou outro IP onde nivela e acessa os filtros (como aqueles provido por um IP-nivel Firewall). É recomendado para servidores contendo informações não públicas.

4.20 CONFIGURANDO O SERVIDOR LDAP

Toda configuração execução do slapd é realizada pelo arquivo slapd.conf, instalado no diretório de prefixo especificado durante o processo de instalação e configuração (compilação se for o caso) ou à revelia em /usr/local/etc/openldap.

Deve-se detalhar a configuração das diretivas usada comumente no slapd.conf. O arquivo de configuração das diretivas são separados em globais, backend específicos e banco de dados específicos. Serão descritas as diretivas junto aos seus valores dafaults para a utilização.

O arquivo slapd.conf consiste em três tipos de informações de configuração: global, backend específico, e banco de dados específico. As informações globais são especificados primeiro, seguidos por informações associadas a um particular tipo de backend, que é então seguidas por informações associadas com uma instância de banco de dados particular.

A diretiva Global pode ser anulada em um backend e/ou diretiva de banco de dados, diretivas backend pode ser anulado por diretivas de banco de dados.

As linhas de comentário brancas começando com um '#' caractere são ignorados. Se uma linha começa com branco especial, é considerada uma continuação da linha prévia (ainda que a linha prévia é um comentário). O formato geral de slapd.conf é:

#configuração global directives

<global config directives>

 # backend definição
backend <typeA>
<backend-specific directives>

#segunda definição de banco de dados & configuração de diretivas
database <typeB>
<database-specific directives>

#segunda "typeA" banco de dados definição & configuração de diretivas
database <typeA>
<database-specific directives>

 # subseqüente backend & definições de banco de dados & configuração de diretivas

A distribuição contém um arquivo de configuração de exemplo que é instalado no diretório /usr/local/etc/openldap. Vários arquivos contendo definições de shemas (tipos de atributo e classes de objeto) também estão providos no diretório /usr/local/etc/openldap/schema.

4.20.1 DIRETIVAS GLOBAIS

Diretivas descritas aqui se aplica a todo backends e bancos de dados a menos que especificamente anulados em um backend ou definição de banco de dados. Os parâmetros que deviam ser substituídos por texto real são mostrados em parênteses <>.

access to <what> [by <who> <accesslevel> <control>]+

Essa diretiva concede acesso (especificado por <accesslevel>) para um conjunto de entradas e/ou atributos por um ou mais requisições.

Se nenhum acesso as diretivas forem especificadas, o acesso default controla a política.

attributetype <RFC2252 Attribute Type Description>

Essa diretiva define um tipo de atributo. Checar o URL seguinte para mais detalhes:

idletimeout <integer>

Especifica o número de segundos para esperar na frente de violentamente fechar uma conexão de cliente inativo. Um idletimeout 0, o default, desativa esta característica.

include <filename>

Essa diretiva especifica que o slapd deve ler informações de configuração adicionais do arquivo dados antes de continuar com a linha próxima do arquivo corrente. O arquivo incluído devia seguir o normal formato do arquivo slapd. O arquivo é comumente usado para incluir arquivos contendo especificações da schema.

loglevel <integer>

Essa diretiva especifica o nível em que deve depurar declarações e estatística de operação. Deve-se ter OpenLDAP configurado --ativar-depura (o default) para este trabalho (com exceção dos dois níveis de estatística, que estão sempre habilitados). Níveis de log são aditivos. Os valores possíveis para <inteiro> são:

Depurando Níveis

Nível	**Descrição**
-1	Ativar toda depuração
0	Nenhuma depuração
1	Traça chamadas de função
2	Depura manipulação de pacote
4	Traça depuração pesada
8	Gerenciamento de conexão
16	Imprime saída de pacotes enviados e recebidos
32	Procura processamento de filtro
64	Processamento de arquivo de configuração
128	Processamento do acesso a lista de controle
256	Ativa log de conexões / operações / resultados.
512	Ativa a entrada no log das enviadas
1024	Comunicação de impressão com shell backends
2048	Imprime analise de depuração da entrada.

Exemplo:

loglevel 255 or loglevel -1

Este causará muitas informações de depuração para ser syslogged.

Default:

loglevel 256

objectclass <RFC2252 Object Class Description>

Essa diretiva define uma classe de objeto. Checar a URL seguinte para mais detalhes.

referral <URI>

Essa diretiva especifica a indicação para passar de volta quando slapd puder encontrar um para o banco de dados um pedido local.

referral ldap://root.mazombo.br

Este se referirá a questões não locais da raiz global do servidor LDAP no Projeto OpenLDAP. os clientes do LDAP podem perguntar em um determinado servidor, mas nota que a maior parte destes clientes só vão conhecer como URLs do LDAP simples que contêm um host separado e opcionalmente um nome distinto.

sizelimit <integer>

Essa diretiva especifica o número de máximo de entradas para retornar para uma operação de busca.
Default:
sizelimit 500

timelimit <integer>

Essa diretiva especifica o número de máximo de segundos (em tempo real) que o slapd passará a responder a um pedido de procura. Se um pedido não está acabado neste tempo, um resultado indicando um excedido timelimit será retornado.
Default:
timelimit 3600

4.20.2 DIRETIVAS GERAL DE BACKEND

Diretivas deste tópico se aplica só para o backend em que eles são definidas. Eles são suportados por todo tipo de backend. Diretivas de backend se aplica a todas as instâncias de bancos de dados do mesmo tipo e, dependendo da diretiva, pode ser anulados por diretivas do banco de dados.

backend <type>

Essa diretivas marcam o início de uma definição de backend. <o tipo> devia ser um de bdb ou um de outro suporta tipos backend listados abaixo:

Banco de dados Backends

Tipo	Descrição
bdb	Berkeley DB transactional backend
dnssrv	DNS SRV backend
ldbm	Lightweight DBM backend
ldap	Lightweight Directory Access Protocol (Proxy) backend
meta	Meta Directory backend
monitor	Monitor backend
passwd	Permissão de somente leitura para o passwd
perl	Perl programmable backend
shell	Shell (programa externo) backend
sql	Programável em SQL backend

Exemplo:
backend bdb

Esta marca o início de um nova definição para o BDB backend.

4.20.3 BANCO DE DADOS GERAL DE DIRETIVAS

Diretivas neste tópico se aplica só para o banco de dados em que eles são definidos. Eles são suportados por todo tipo de banco de dados.

database <type>

Esta diretiva marca o início de uma nova definição de instância de banco de dados.

database bdb

Esta marca o início de um nova BDB backend definição da instância de banco de dados.

readonly { on | off }

Esta diretiva põe o banco de dados no modo "somente para leitura". Qualquer tentativas para modificar o banco de dados retornará um mensagem de erro.
Default:
readonly off

replica uri=ldap[s]://<hostname>[:<port>] |
host=<hostname>[:<port>]
[bindmethod={simple|kerberos|sasl}]
["binddn=<DN>"]
[saslmech=<mech>]
[authcid=<identity>]
[authzid=<identity>]
[credentials=<password>]
[srvtab=<filename>]

Esta diretiva especifica uma replicação do site para este banco de dados. O uri= parâmetro especifica um esquema, um host e opcionalmente uma porta onde o escravo slapd instância e pode ser achada. Ou um nome de domínio ou endereço de IP podem ser usados para <hostname>. Se <porta> não recebe, o número de porta de padrão LDAP (389 ou 636) é usado.

A uri permite o servidor de réplica LDAP ser especificado como um LDAP URI como ldap://slave.mazombo.br:389 ou ldaps://slave.mazombo.br:636.

O binddn é o parâmetro que dá ao DN a vincular como para atualizações para o escravo slapd. Devia ser um DN que tem acesso *read/write* ao banco de dados escravo slapd. Deve também combinar a diretiva updatedn no arquivo de configuração slapd do escravo. Geralmente, este DN *não devia* ser o mesmo que o rootdn no banco de dados mestre. Desde que os DNS são prováveis para conter espaços embutidos, os inteiros "binddn=<DN>" string deve ser incluso em citações duplas.

O bindmethod é simples , pode usar kerberos ou sasl, dependendo da autenticação baseada em senha simples ou autenticação de Kerberos ou autenticação de SASL é para ser usada quando conectando ao escravo slapd.

A autenticação simples não devia ser usada a menos que integridade e proteções de isolamento adequadas estão em lugar (por exemplo TLS ou IPSEC). Autenticação simples exige especificação de binddn e parâmetros credenciais.

A autenticação de Kerberos é voltada a favor de mecanismos de autenticação de SASL, em particular os mecanismos de KERBEROS_V4 e GSSAPI. a autenticação de Kerberos exige binddn e srvtab parâmetros.

A autenticação de SASL está geralmente recomendada. a autenticação de SASL exige especificação de um mecanismo usando o parâmetro saslmech. Dependendo do mecanismo, uma identidade de autenticação e/ou credenciais podem ser especificadas usando authcid

e credenciais respectivamente. O authzid parâmetro pode ser usado para especificar uma identidade de autorização.

replogfile <filename>

Essa diretiva especifica o nome do arquivo log da replicação para o slapd. O log da replicação é tipicamente escrito por slapd e lê por slurpd. Normalmente, esta diretiva é só usada se slurpd está sendo usado para reproduzir o banco de dados. Porém, também pode usar ele para gerar um log de transação, se slurpd não está executando. Neste caso, periodicamente precisará truncar o arquivo, caso contrário ele crescerá indefinidamente.

rootdn <dn>

Esta diretiva especifica o DN que não é sujeito a controle de acesso ou restrições de limite administrativas para operações neste banco de dados. O DN não precisa se refere a uma entrada no diretório. O DN pode se referir a uma identidade de SASL.

Exemplo baseado em entrada:
rootdn "cn=Manager, dc=mazombo, dc=br"

Exemplo baseado em SASL:
rootdn "uid=root,cn=mazombo.br,cn=digest-md5,cn=auth"

rootpw <password>

Esta diretiva pode ser usada para especificar uma senha para o rootdn (quando o rootdn é configurado para um DN dentro do banco de dados).
Exemplo:
rootpw secret

Também é permissíveis para prover o formato da senha no formulário RFC 2307. slappasswd pode ser usado para gerar a segurança da senha.
Exemplo:
rootpw {SSHA}ZKKuqbEKJfKSXyfgftrtbHG3fn9j1v4QNG8MD

O código foi gerado usando o comando slappasswd -s secret.

suffix <dn suffix>

Esta diretiva especifica o sufixo de DN de questões que serão passadas para este backend banco de dados. As linhas de sufixo múltiplo podem receber, e pelo menos se é exigido para cada definição de banco de dados.

suffix "dc=mazombo, dc=br"

As questões com um DN terminar em "dc=mazombo.dc=br" será passado para este backend.

Quando o backend para passar uma questão é selecionado, slapd dirige-se para a linha de sufixo(s) em cada definição de banco de dados no pedido eles aparecem no arquivo. Deste modo, se um sufixo de banco de dados é um prefixo de outro, deve aparecer atrás dele no config arquivo.

Syncrepl

Esta diretiva é usado para manter um banco de dados reproduzido sincronizado com o banco de dados de mestre, de forma que o conteúdo de banco de dados reproduzido será mantido em dia com o conteúdo do mestre.

Neste trabalho não cobre em detalhes este diretivo, porque esta sendo configurando um Servidor de LDAP único.

updatedn <dn>

Esta diretiva é só aplicável em um escravo slapd. Especifica o DN permitiu fazer mudanças para a réplica. Isto pode ser o DN slurpd vincula como quando fazendo mudanças para a réplica ou o DN associou com uma identidade de SASL.

Exemplo baseado em entrada:
updatedn "cn=Update Daemon, dc=mazombo, dc=br"

Exemplo baseado em SASL:
updatedn "uid=slurpd,cn=mazombo.br,cn=digest-md5,cn=auth"

updateref <URL>

Esta diretiva é só aplicável em um escravo slapd. Especifica-se a URL para retornar aos clientes que se submetem a pedidos de atualização na réplica.

updateref ldap://master.mazombo.br

4.20.4 BDB BANCO DE DADOS DE DIRETIVAS

Diretivas nesta categoria só se aplica um banco de dados de BDB. Isto é, eles devem seguir um "banco de dados linha bdb", vem antes de qualquer subseqüente "backend" ou "linha de banco de dados.

directory <directory>

Esta diretiva especifica o diretório onde os arquivos de BDB contendo o banco de dados e associou índices.
Default:
directory /usr/local/var/openldap-data

sessionlog <sid> <limit>

A operação de Sincronização de Conteúdo de LDAP em pré-existindo sessão pode usar o arquivo de log de sessão a fim de reduzir a quantia de tráfego de sincronização. Se a réplica não é tão antiquada que pode ser feito em dia pelas informações no arquivo de sessão, o provedor slapd enviará ao consumidor slapd as identidades para entradas junto com as entradas de escopo adicionado as modificações dentro do conteúdo da replicação. Se o status de réplica é antiquado demais e além da cobertura do arquivo do histórico, então o provedor slapd enviará as identidades das descarregadas em entradas de escopo junto com as mudadas entradas de escopo. O consumidor slapd então removerá entradas na réplica que não são identificadas como apresentados no conteúdo de provedor.

4.20.5 LDBM BANCO DE DADOS DE DIRETIVAS

Diretivas nesta categoria só se aplica ao banco de dados LDBM backend. Isto é, eles devem seguir um "banco de dados ldbm" e vem antes de qualquer outro "banco de dados" ou "backend".

cachesize <integer>

Esta diretiva especifica o tamanho das entradas do em cache de memória mantida pela instância do banco de dados LDBM backend.

Default:

cachesize 1000

dbcachesize <integer>

Esta diretiva especifica o tamanho em bytes do cache de memória associada com cada arquivo de índice aberto. Se não suportado pelo método de banco de dados subjacente, a diretiva é ignorada sem comentário. Acrescentando este número usa mais memória mas pode fazer um aumento de apresentação dramática, especialmente durante modificação ou quando constroí índices.

Default:

dbcachesize 100000

dbnolocking

Esta opção, se presente, desativa bloqueio de banco de dados. Ativando esta opção pode melhorar apresentação às custas de segurança de dados.

Dbnosync

Esta causa a opção do conteúdo de banco de dados do disco sem ser imediatamente sincronizadas com mudanças da memória. Ativando esta opção pode melhorar apresentação às custas de segurança de dados.

directory <directory>

Esta diretiva especifica o diretório onde os arquivos de LDBM contendo o banco de dados associou seus índices.

Default:

directory /usr/local/var/openldap-data

index {<attrlist> | default} [pres,eq,approx,sub,none]

Esta diretiva especifica os índices para manter o atributo dado. Se só um <attrlist> recebe, os índices defaults são mantidos.

index default pres,eq
index uid
index cn,sn pres,eq,sub
index objectClass eq

A primeira linha fixa o conjunto default de índices para manter presente e igual. A segunda linha causa o default (pres,eq) conjunto de índices para ser mantido para o atributo uid. A linha três apresenta, igualdade e índices de substring para ser mantidos para atributo cn e sn. A quarta linha causa um índice de igualdade para o atributo objectClass.

index objectClass eq

mode <integer>

Esta diretiva especifica o modo de proteção de arquivos recentes do índice de banco de dados criados.
Default:
mode 0600

4.20.6 CONTROLE DE ACESSOS

O acesso controla instalação provida pelo acesso diretivo é bastante poderoso. Esta seção frisa demonstrações de uso.

access to * by * read

É demonstrado o uso de uma expressão regular para selecionar as entradas por DN e acessa diretivas onde ordenar é significante.

access to dn=".*, o=mazombo, c=BR"
by * search
access to dn=".*, c=BR"
by * read

Acesso é concedido para entradas debaixo da subárvore de c=BR, com exceção das entradas debaixo das "o=mazombo, c=BR" subárvore, para procurar acesso concedido. Nenhum acesso é concedido para c=BR, como nenhuma partidas de acesso diretivas DN. Se o pedido destes acessos a diretivas eram invertidos. Outro caminho para implementar os mesmos controles de acesso é:

```
access to dn.amigos="dc=mazombo,dc=br"
        by * search
        access to dn.amigos="dc=br"
        by * read
```

Acesso é concedido para entradas debaixo da subárvore de “br” de dc, com exceção das entradas debaixo da subárvore dc=mazombo,dc=br, para que procurem acesso concedido. Nenhum acesso é concedido para dc=br ,nenhuma partida do acesso da diretivas DN. Se o pedido destes acesso a diretivas eram invertidos, arrastar diretiva nunca seria alcançado, desde que todas as entradas debaixo de dc=mazombo,dc=br, também estão debaixo de entradas de dec=br.

Também nota-se nenhum acesso para cláusula da diretiva "by <que>", o acesso é negado. Isto é, todo *acesso a* diretiva implícita *by * none* cláusula e todos acessam a lista, termina com um acesso implícito a diretiva ** by * none*.

O próximo arquivo de configuração, novamente mostra à importância de ordenar, ambos acesso a cláusulas da diretiva "by <que>". Também mostra o uso de um seletor de atributo para conceder acesso a um atributo específico e vários <que> seletores.

```
access to dn.subarvore="dc=mazombo,dc=br" attr=telefones
        by self write
        by dn.amigos=dc=mazombo,dc=br" search
        by peername=IP:10\..+ read
access to dn.subarvore="dc=mazombo,dc=br"
        by self write
        by dn.amigos="dc=mazombo,dc=br" search
        by anonymous auth
```

Esta demonstração se aplica a entradas nas subárvore "dc=mazombo,dc=br". Todos os atributos exceto “telefone”, uma entrada pode escrever para propriamente, entradas debaixo de mazombo.br entradas podem procurar por eles, qualquer outra pessoa não tem nenhum acesso (implícito by * none) com exceção de autenticação/autorização (que é sempre feito anonimamente). O atributo “telefone” é escrita pela entrada, procura por entradas debaixo de mazombo.br, legíveis por clientes que conectam na faixa de rede 10, caso contrário não legível (implícita by * none). Todo outro acesso é negado pelo acesso implícito a * by * none.

Às vezes é útil para permitir um DN particular para adicionar ou remover propriamente um atributo.se quiser criar um agrupamento e permitir a pessoas adicionar e remover seu próprio atributo DN do membro, podie realizar com um acesso diretivo como isto:

access to attr=member,entry

by dnattr=member selfwrite

O dnattr diz que o acesso se aplica a entradas listadas no atributo do membro. O selfwrite acessa diz que tais membros podem adicionar ou apagar seu próprio atributo DN. A adição do atributo de entrada é exigida porque o acesso a entrada é exigido para acessar quaisquer dos atributos de entrada.

4.20.7 ARQUIVO DE CONFIGURAÇÃO

O seguinte é um arquivo de configuração, com texto explicativo. Define partes de bancos de dados diferentes da árvore de X.500; ambos são instâncias de banco de dados de BDB. O números das linhas mostrados são providos para referência e não são incluídas no arquivo real. Primeiro, a seção de configuração global:

```
1.  # config file - global configuration section
2.  include /usr/local/etc/schema/core.schema
3.  referral ldap://root.openldap.org
4.  access to * by * read
```

A linha 4 é um controle de acesso global. Se aplica a todas as entradas.

A próxima seção do arquivo de configuração define questões para parâmetros, "dc=mazombo,dc=br" da árvore. O banco de dados é para ser reproduzido para dois escravo slapds. Os índices são para ser mantidos para vários atributos, e o atributo userPassword é para ser protegido de acesso sem autorização.

```
5.  # BDB definition for the mazombo.com
6.  database bdb
7.  suffix "dc=mazombo,dc=br"
8.  directory /usr/local/var/openldap-data
9.  rootdn "cn=Manager,dc=mazombo,dc=br"
10.  rootpw secret
11.  # replication directives
12.  replogfile /usr/local/var/openldap/slapd.replog
13.     replica uri=ldap://slave1.mazombo.br:389
14.        binddn="cn=Replicator,dc=mazombo,dc=br"
15.        bindmethod=simple credentials=secret
16.  replica uri=ldaps://slave2.mazombo.br:636
17.        binddn="cn=Replicator,dc=mazombo,dc=br"
18.        bindmethod=simple credentials=secret
19.  # indexed attribute definitions
```

```
20.  index uid pres,eq
21.  index cn,sn,uid pres,eq,sub
22.  index objectClass eq
23.  # database access control definitions
24.  access to attr=userPassword
25.          by self write
26.          by anonymous auth
27.          by dn.base="cn=Admin,dc=mazombo,dc=br" write
28.          by * none
29.  access to *
30.          by self write
31.          by dn.base="cn=Admin,dc=mazombo,dc=br" write
32.          by * read
```

A linha 5 é um comentário. O começo da definição de banco de dados é marcado pelo palavra-chave de banco de dados na linha 6. A linha 7 especifica o sufixo de DN para questões passar para este banco de dados. A linha 8 especifica o diretório em que os arquivos de banco de dados ficarão.

As linhas 9 e 10 identificam o banco de dados " super usuário", entrada e associa a senha. Esta entrada não é sujeito a controle de acesso, tamanho ou restrições de prazo.

```
rootpw {SSHA}ZKKuqbEKJfKSXyfgftrtbHG3fn9j1v4QNG8MD
```

As linhas 11 a 18 são para replicação.

As linhas 20 a 22 indicam os índices para manter vários atributos.

As linhas 24 por 32 especificam controle de acesso para entradas no banco de dados. Como é o primeiro banco de dados, os controles também se aplicam a entradas não seguras em qualquer banco de dados. Para todas as entradas aplicáveis, o userPassword atributo é escrita pela entrada propriamente e pela entrada "admin". Pode ser usado para autenticação e/ou autorização, mas caso contrário não é legível. Todos outros atributos são escritos pela entrada "admin", mas pode ser lida por todos os usuários (autenticado ou não).

A próxima seção do arquivo de configuração define outro banco de dados de BDB. Aqui esta questões envolvendo o dc=mazombo,dc=br, é administrada pela mesma entidade que o primeiro banco de dados.

```
33.  # BDB definition for mazombo.net
34.  database bdb
35.  suffix "dc=mazombo,dc=net"
36.  directory /usr/local/var/openldap-data-net
37.  rootdn "cn=Manager,dc=mazombo,dc=br"
38.  index objectClass eq
39.          access to * by users read
```

4.20.8 EXECUÇÃO DO SERVIDOR LDAP

O processo de LDAP *slapd* é projetado para ser executado como um servidor independente. Este permite o servidor para aproveitar o caching e adminitração corrente dos assuntos com bancos de dados subjacentes, e conservam recursos de sistema.

4.20.9 OPÇÕES DE LINHAS DE COMANDOS

Slapd suporta várias opções de linha de comandos.

-f <filename>

Esta opção especifica um arquivo de configuração alternada para slapd. O default é normalmente /usr/local/etc/openldap/slapd.conf.

-h <URLs>

Esta opção especifica configurações de localização alternativo. O default é ldap:/// que implica LDAP acima de TCP em todas as interfaces na porta do LDAP default 389. Pode especificar um host específicos ou outros esquemas de protocolo (como ldaps:// ou ldapi:///).

-n <service-name>

Esta opção especifica o nome de serviço usado para logging e outros propósitos. O nome de serviço default é slapd.

-l <syslog-local-user>

Esta opção especifica o usuário local para instalação syslog. Os valores podem ser LOCAL0, LOCAL1, LOCAL2,..., e LOCAL7. O default é LOCAL4. Esta opção não pode ser suportada em todos os sistemas.

-u user -g group

Estas opções especificam o usuário e grupos, respectivamente. O usuário pode ser um nome do usuário ou uid.Grupo pode ser nome de grupo ou gid.

-r directory

Esta opção especifica tempo de execução do diretório. Slapd, chroot para este diretório depois de aberto escuta, antes de ler qualquer arquivo de configuração ou inicializar qualquer backends.

-d <level> | ?

Esta opção fixa para o slapd depurar níveis. Caracteristicas dos vários níveis de depuração estão impressos e slapd, não importando quaisquer outras opções e os níveis de depuração corrente:
Pode ativar níveis múltiplos especificando a opção depurar, uma vez para cada nível. Isto é, se quiser localizar função chamada e verificação no arquivo de configuração, estando processada, pode configurar nível para a soma daqueles dois níveis (neste caso, -d 65). Ou, pode deixar slapd fazer, (por exemplo -d 1 -d 64).

4.20.10 CRIAÇÃO DE BANCO DE DADOS E MANUTENÇÃO

Esta seção especifica como criar um banco de dados slapd. Existem dois caminhos para criar um banco de dados. Primeiro pode criar o banco de dados on-line usando LDAP. Com este método, simplesmente recomeça atividades slapd e adiciona entradas usando o cliente de LDAP da escolha do administrador. Este método é bom para bancos de dados relativamente pequenos (algumas cem ou mil entradas, dependendo de seus requisitos). Este trabalho de método para tipos de banco de dados suporta atualizações.

O segundo método de criação de banco de dados está para fazer fora da linha usando utilitários especiais provido com slapd. Este método é melhor se você tiver muitos milhares de entradas para criar, que tomariam um tempo longo usando o método de LDAP, ou se quiser assegurar o banco de dados não é acessado enquanto está sendo criado. Só que nem todos tipos de banco de dados suportar estes utilitários.

4.20.11 CRIANDO BANCO DE DADOS ON-LINE

O pacote de software de OpenLDAP vem com um utilitário chamado ldapadd, costuma adicionar entradas enquanto o servidor de LDAP está em execução. Se optar por criar o Banco de dados online, pode usar o ldapadd ferramenta para adicionar entradas (também pode usar outros clientes providos do pacote de OpenLDAP adicionar entradas). Depois de adicionar as primeiras entradas, pode ainda usar ldapadd para adicionar mais entradas. Não deve deixar de configurar as opções de configuração seguinte no arquivo sladp.conf antes de começar slapd:

suffix <dn>

Esta opção diz quais entradas são para ser seguras por este banco de dados. Deve-se deixar este DN na raiz da subárvore que está tentando criar.

suffix "o=mazombo, c=br"

Não deve deixar de especificar um diretório onde os arquivos de índice devem ser criados:

directory /usr/local/tudelft

Este diretório e necessário ser criado com permissões apropriadas de forma que slapd possa escrever.

E preciso configurar slapd de forma que possa conectar a ele como um usuário de diretório com permissão para adicionar entradas. Configurar o diretório para suportar um usuário ou usuário de raiz. Isso é feito pelas seguintes opções na definição de banco de dados:

rootdn <dn>

rootpw <passwd> (Utiliza-se uma senha de SHA)

Estas opções especificam um DN e senha que podem ser usados para autenticar como o "superusuario" entrada do banco de dados (isto é, a entrada permiti fazer qualquer coisa). O DN e senha especificaram aqui sempre trabalharão, não importando se a entrada chamado realmente existe ou tem a senha. Este resolve o problema de como autenticar e adicionar entradas antes de qualquer entrada.

Slapd nativo entende se usar um SHA-1 codificando a senha na diretiva rootpw. É possível usar o comando *slappasswd* para gerar as senhas.

slappasswd -h {SHA}

rootpw "{SHA}5en6G6MezRroT3XKqkdPOmY/BfQ="

rootdn "cn=Manager,dc=mazombo,dc=br"

rootpw "{SHA}5en6G6MezRroT3XKqkdPOmY/BfQ="

A saída default para slappasswd é para gerar senhas seguras {SSHA}, neste caso não precisa passar pelo parâmetro -h, chama-se slappasswd diretamente.

Se estiver usando SASL como um mecanismo para autenticar contra LDAP, a linha rootpw pode ser descartada.

Finalmente, deve ter certeza que a definição de banco de dados contenha as definições de índice que deseja:

index {<attrlist> | default} [pres,eq,sub,none]

Índice dos atributos cn, sn, uid e objectclass, as linhas de configuração de índice seguinte puderam ser usadas.

index cn,sn,uid pres,eq,sub

index objectClass pres,eq

Não são todos tipos de índice que estão disponíveis com todos tipos de atributo. Uma vez que você configurou para sua preferência, recomece atividades slapd, conecte com seu cliente de LDAP, e comece a adicionar entradas. Por exemplo, adicionar a entrada de mazombo seguido por uma entrada de “email” usando o ldapadd ferramenta, cria um arquivo chamado /tmp/novaentrada com o conteúdo:

o=mazombo, c=br
objectClass=organization
description=Teste do dominio mazombo

cn=email, o=mazombo, c=br
objectClass=organizationalRole
cn=email
description= mazombo email – email@mazombo.br

Então usa-se um comando para criar a entrada:

ldapadd -f /tmp/novaentrada -x -D "cn=Manager, o=mazombo, c=br" -w secret

O comando acima assume que foi configurado rootdn "cn=Gerente, o=mazombo, c=br" e rootpw para "secreto" (talvez SHA-1 codificado em slapd.conf). Se não quiser a senha na linha de comandos, usa-se a opção -W para o comando ldapadd em vez de -w "senha".

ldapadd -f /tmp/newentry -x -D "cn=Manager, o=mazombo, c=br" -W

Enter LDAP Password:

4.20.12 CRIANDO UM BANCO DE DADOS OFFLINE

O segundo método de criação de banco de dados faz isto fora da linha, usando o slapd, banco de dados e ferramentas descritas abaixo. Este método é melhor se tiver muitos milhares de entradas para criar, que tomariam um tempo longo usando o método e ferramentas do LDAP. Estas ferramentas lêem o arquivo de configuração slapd e um arquivo de entrada LDIF contendo uma representação em texto das entradas para serem adicionadas. Para tipos de banco de dados que suportam as ferramentas, eles produzem os arquivos de banco de dados diretamente (caso contrário deve usar o método on-line). Existem várias opções de configuração importante que estará certo e aparece a definição do banco de dados de arquivo primeiro:

suffix <dn>

Esta opção diz que entradas são para ser seguras por este banco de dados. Devie-se deixar este para o DN da raiz da subárvore que está tentando criar.

suffix "o=mazombo, c=br"

Não deve deixar de especificar um diretório onde os arquivos de índice devam ser criados:

directory /usr/local/mazombo

Precisa especificar quais índices deseja construir. É feito por uma ou mais opções de índice.

index {<attrlist> | default } [pres,eq,approx,sub,none]

index cn,sn,uid pres,eq,sub
index objectClass eq

Este cria presença, igualdade e índices de substring para o cn, sn, e uid atributos e um índice de igualdade para o atributo objectClass. Uma vez que configurado de acordo com a preferência do adminitrador, cria um banco de dados primário e associa índices executando o programa slapadd:

slapadd -l <inputfile> -f <slapdconfigfile> [-d <debuglevel>] [-n <integer>|-b <suffix>]

Os parâmetros têm os seguintes significados:

-l <inputfile>

O arquivo de entrada e especificado no LDIF contendo as entradas para adicionar em formulário de texto.

-f <slapdconfigfile>

Especifica-se o arquivo de configuração slapd que diz onde criar os índices.

-d <debuglevel>

Aciona a depuração, como especificados por <debuglevel>. O depurar níveis são o mesmo que para slapd.

-n <databasenumber>

Um parâmetro opcional que especifica qual banco de dados deve modificar. O primeiro banco de dados listado no arquivo de configuração é 1, o segundo 2, etc. À revelia, o primeiro banco de

dados no arquivo de configuração é usada. Não devia ser usado junto com -B.

-b <suffix>

Um parâmetro opcional que especifica o que banco de dados deve modificar. O sufixo provido é combinado contra um sufixo de banco de dados diretivo para determinar o número de banco de dados. Não deve ser usado junto com -N.

Às vezes pode ser necessário para regenerar índices, como depois de modificar slapd.conf. Isto é possível usando o programa slapindex.

slapindex -f <slapdconfigfile> [-d <debuglevel>] [-n <databasenumber>|-b <suffix>]

Onde as -f, -d, -n e -b são opções o mesmo que para o slapadd programa. slapindex reconstrói todos os índices baseados no conteúdo de banco de dados corrente.

O programa slapcat é usado para esvaziar o banco de dados para um arquivo de LDIF. Isso pode ser útil quando quiser fazer uma cópia de segurança legível de seu banco de dados ou quando quiser editar seu banco de dados fora da linha.

slapcat -l <filename> -f <slapdconfigfile> [-d <debuglevel>] [-n <databasenumber>|-b <suffix>]

Onde -n ou -b é usado para selecionar o banco de dados no slapd.conf, especificado usando -F. A saída de LDIF correspondente é escrita para saída de padrão ou para o arquivo especificado usando a opção -l.

4.20.13 FORMATO DO LDIF

O formato de intercâmbio de Dados dO LDAP (LDIF) é usado para representar entradas de LDAP em um formato de texto simples. O formulário básico de uma entrada é:

#comment
dn: <distinguished name>
<attrdesc>: <attrvalue>
<attrdesc>: <attrvalue>
...

As linhas que começam com um '#' caractere são comentários. Uma descrição de atributo (attrdesc) pode ser um tipo de atributo simples como cn ou objectClass ou podem incluir opções como cn;lang_en_US ou userCertificate;binary.

dn: cn=Barbara Silva, dc=mazombo, dc=br
cn: Barbara Silva

#este equivale a
dn: cn= Barbara Silva, dc=mazombo, dc=br
cn: Barbara Silva

#os valores de atributo múltiplo são especificados nas linhas separadas.
cn: Barbara Silva
cn: Barbarinha

Se um <attrvalue> contém caracteres não-imprimindo ou começar com um especial, um dois pontos duplo (':'), ou um menos que ('<'), o <attrdesc> é seguido por um dois pontos duplo e a base64 de codificação do valor.

cn:: IGJlZ2lucyB3aXRoIGEgc3BhY2U=

Também pode especificar uma URL contendo o valor de atributo.

cn:< file://path/to/file.jpeg

As entradas múltiplas dentro do mesmo arquivo de LDIF são separadas pelas linhas brancas. Aqui é um arquivo de LDIF contendo duas entradas.

 # Entrada de Jose Silva
dn: cn=Jose da Silva, dc=mazombo, dc=br
cn: Jose da Silva
cn: Ze da Silva
objectClass: person
sn: Silva

 # Base64 encoded JPEG imagens
jpegPhoto:: /9j/4AAQSkZJRgABAAAAAQABAAD/2wBDABALD
A4MChAODQ4SERATGCgaGBYWGDEjJR0oOjM9PDkzODdASFxO
Q
ERXRTc4UG1RV19iZ2hnPk1xeXBkeFxlZ2P/2wBDARESEhgVG

 # Entrada de Antonio Souza
dn: cn=Antonio Souza, dc=mazombo, dc=com
cn: Antonio Souza
cn: Antonio Souza
objectClass: person
sn: Souza
 # arquivos de imagens .jpeg
imagensjpeg:< file://srv/arquivo.jpeg

Espaços não são aparados de valores em um arquivo de LDIF. Não são espaços internos múltiplos comprimidos. Se não quiser eles em seus dados,e só não coloca-los.

4.20.14 O LDAPSEARCH, LDAPDELETE E LDAPMODIFY

ldapsearch é uma interface do shell acessível para o ldap_search, chamando sua biblioteca. Usa-se para procura de utilitário para entradas em seu banco de dados de LDAP.

A sinopse para chamar ldapsearch é o seguinte:

ldapsearch [-n] [-u] [-v] [-k]
[-K] [-t] [-A] [-B] [-L]
[-R] [-d debuglevel] [-F sep] [-f file]
[-x] [-D binddn] [-W] [-w bindpasswd]
[-h ldaphost] [-p ldapport] [-b searchbase]
[-s base|one|sub]
[-a never|always|search|find] [-l timelimit]
[-z sizelimit] filter [attrs...]

Apresenta uma procura usando o filtro. O filtro deve ajustar com a representação de string para LDAP definido em RFC 1558. Se ldapsearch acha um ou mais entradas, os atributos especificados por *attrs* são recuperados e as entradas e valores estarão impressos para saída de padrão. Se nenhum attrs forem listados, todos os atributos são retornados.

ldapsearch -x -b 'o=mazombo,c=br' 'objectclass=*'

ldapsearch -b 'o= mazombo,c=br ' 'cn=Jose Silva'

ldasearch -u -b 'o= mazombo,c=br ' 'cn=Antonio Souza' sn mail

A opção -b suporta searchbase (ponto de procura inicial), opção a -u suportam userfriendly saída de informações e a opção -x é usado para especificar autenticação simples.

ldapdelete é uma interface do shell acessível para o ldap delete chamando sua biblioteca. Use este utilitário para apagar entradas em nosso banco de dados de LDAP.

A sinopse para chamar ldapdelete é o:

ldapdelete [-n] [-v] [-k] [-K]
[-c] [-d debuglevel] [-f file] [-D binddn]
[-W] [-w passwd] [-h ldaphost] [-p ldapport]
[dn]...

ldapdelete abre uma conexão para um servidor de LDAP, vincula, e apaga uma ou mais entradas. Se um ou mais parâmetros dn são providos, de entradas com aqueles Nomes Distintos são apagadas. Cada dn devia ser uma string-representada DN como definido em RFC 1779. Se nenhum parâmetros de dn forem providos, uma lista de DNS é lida na entrada padrão.
ldapdelete 'cn=Jose Silva,o=mazombo,c=br'.

ldapdelete -v 'cn=Antonio Souza,o=mazombo,c=br' -D 'cn=Jose Silva,o=mazombo,c=br' –W

A -v opção suporta modo verboso, a opção -D suporta Binddn (o dn para autenticar contra) e a opção -W suporta lembrete de senha.
ldapmodify é uma interface do shell acessível para o ldap_modifye ldap_add chamando suas bibliotecas. Use este utilitário para modificar entradas em nosso banco de dados de LDAP.
A sinopse para chamar ldapmodify é o seguinte:

ldapmodify [-a] [-b] [-c] [-r]
[-n] [-v] [-k] [-d debuglevel]
[-D binddn] [-W] [-w passwd]
[-h ldaphost] [-p ldapport] [-f file]

ldapadd [-b] [-c] [-r] [-n]
[-v] [-k] [-K] [-d debuglevel]
[-D binddn] [-w passwd] [-h ldaphost]
[-p ldapport] [-f file]

ldapadd é implementado como um link para a ferramenta ldapmodify. Quando invocou como ldapadd a tarefa de adicionar nova flag de entrada o ldapmodify é automaticamente ligado. ldapmodify abre uma conexão para um servidor de LDAP, vincula, e modifica ou adiciona entradas. As informações de entrada são lidas na entrada padrão ou de arquivo pelo uso da opção -f.

dn: cn=modifica,o=mazombo,c=br
changetype: modifica
replace: mail
mail: modif@mazombo.br
-
add: title
title: Principal
-
add: imagensjpeg
imagensjpeg: /tmp/imagens.jpeg
-
delete: description
o comando
ldapmodify -b -r -f /tmp/novaentrada

Substituirá o conteúdo do "modifica" atributo de entrada do correio com o valor " modif@mazombo.br ", adiciona um título "Principal", e o conteúdo do arquivo /tmp/imagens.jpeg como uma imagensjpeg, e completamente remova o atributo de descrição.

As mesmas modificações de acima podem ser apresentadas usando a mais velho formato de entrada ldapmodify:

cn=modifica,o=mazombo,c=br
mail=modif@mazombo.br

+title=principal
+imagensjpeg=/tmp/imagens.jpeg
-description
O comando:
ldapmodify -b -r -f /tmp/entrymods
O arquivo /tmp/novaentrada existe e tem conteúdo:
dn: cn=Maria Silva,o=mazombo,c=br
objectClass: person
cn: Maria Silva
cn: Maie
sn: Silva
title: gerente de marketing
mail: maria.silva@mazombo.br
uid: maria.silva

o comando

ldapadd -f /tmp/entrymods

Adicionará a entrada com dn: cn=Maria Silva, o=mazombo, c=br , se não for já apresenta uma entrada com este dn já existe, o comando assinala o erro e não escrever elaboradamente a entrada.

O arquivo /tmp/novaentrada existe e tem o conteúdo:
dn: cn=Maria Silva,o=mazombo,c=br

changetype: delete

o comando
ldapmodify -f /tmp/novaentrada
remove a entrada da Maria Silva

A opção -f suporta arquivo, a opção -b suporta binário, o -r suporta substituir valores existentes à revelia.

4.21 AUTENTICAÇÃO USANDO LDAP

Para acessar o serviço de LDAP, o cliente primeiro deve autentica-se no serviço. Isto é, deve dizer ao servidor de LDAP que vai estar acessando os dados de forma que o servidor pode decidir o que o cliente tem permissão para ver e fazer. Se o cliente autentica com sucesso, o servidor subseqüentemente receber um pedido do cliente, checará se o cliente tem permissão para solicitar o pedido. Este processo é chamado controle de acesso.

No LDAP, a autenticação é fornecida na operação “bind”(vincular). Ldapv3 suporta três tipos de autenticação: anônima, simples e autenticação de SASL. Um cliente que envia uma solicitação sem fazer uma vinculação é tratado como um cliente anônimo. A autenticação simples consiste em enviar o servidor de LDAP o completamente DN qualificado do cliente (usuário) e a senha de texto claro do cliente. Este mecanismo tem problemas de segurança porque a senha pode ser lida da rede. Para evitar expor a senha deste modo,

pode-se usar o mecanismo de autenticação simples dentro de um canal codificado (como SSL), desde que suportado pelo servidor de LDAP.

O SASL é a autenticação e a camada de segurança(RFC 2222). Especifica um protocolo em que os dados são permutados entre o cliente e o servidor para os propósitos de autenticação e estabelecimento de uma camada de segurança em que possa executar a comunicação subseqüente. Usando SASL, LDAP pode suportar qualquer tipo de autenticação em acordo estipulado entre o cliente e o servidor.

O processo de autenticação dos usuários acessa informações da sua árvore de diretório, O servidor LDAP pode autenticar usuários de outros serviços (Postifix, ProFtp etc...). Isto é realizado migrando informações de usuário específicas para seu servidor de LDAP e usando um mecanismo chamado PAM, (Pluggable Authentication Module).

4.22 LOGS

Afim de ativar a geração de logs tem que editar o arquivo syslog.conf, localizado no diretório /etc. Cria-se uma linha como a descrita abaixo dentro do arquivo syslog.conf.

local4.* /var/log/slapd.log

Nesta linha usuário default LOCAL4 usará o syslog para facilitar o seu uso. Se quiser especificar o nível dos logs que serão gerados ou mudar o usuário default, existem as opções seguintes quando inicia o slapd:

-s syslog-level

Esta opção diz ao slapd que as declarações de depuração de nível devem ser registradas no syslog. O nível descreve a severidade da mensagem, e é um palavra-chave da seguinte lista ordenada (mais alta para a mais baixa): emerg, alert, crit, err, warning, notice, info, e debug.

slapd -f myslapd.conf -s debug

-l syslog-local-user

Seleciona o usuário local do syslog. Os valores podem ser LOCAL0, LOCAL1, e assim por diante, até LOCAL7. O default é LOCAL4. Porém, esta opção só está permitida em sistemas que suportam usuários local com o syslog.

5. RESULTADO DO PDC CONFIGURADO

Após a verificação das questões teóricas para correta implementação e configuração do servidor PDC utilizando Samba integrado com o OpenLDAP, o resultado será demonstrado através de imagens capturadas de arquivos de configuração e do gerenciamento da base de dados do servidor de diretórios.

5.1 CONFIGURAÇÃO DO SAMBA

A figura 5.1 mostra as configurações iniciais que serão interpretadas pelo servidor samba. Onde constam o domínio, nome do servidor, e configurações com os parâmetros estipulados de acordo com a conveniência e utilidade do momento da configuração.

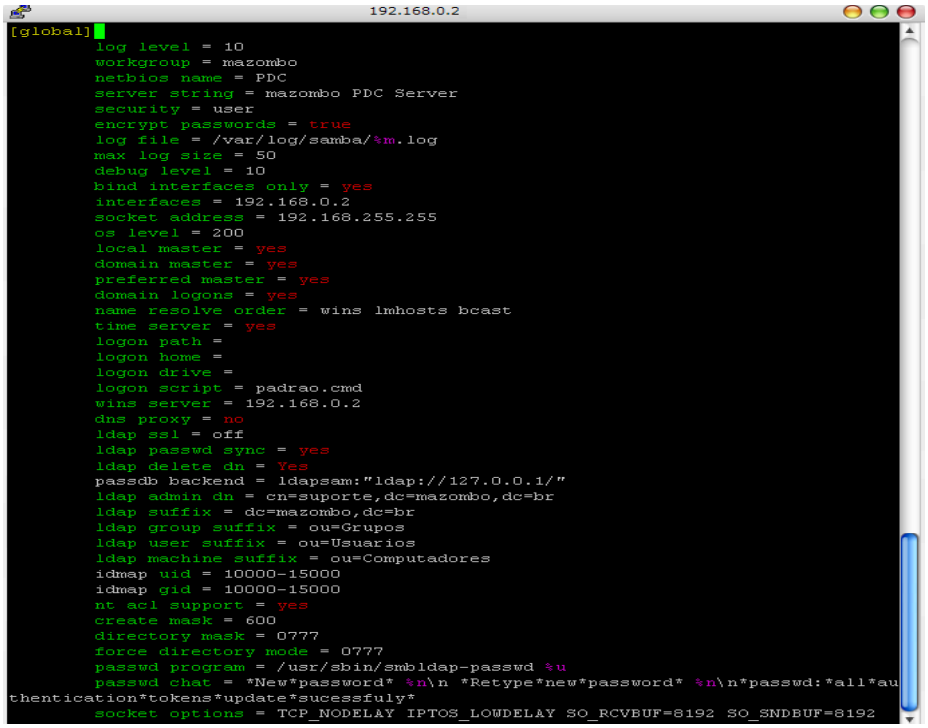

Figura 5.1.1: Primeira parte do smb.conf.

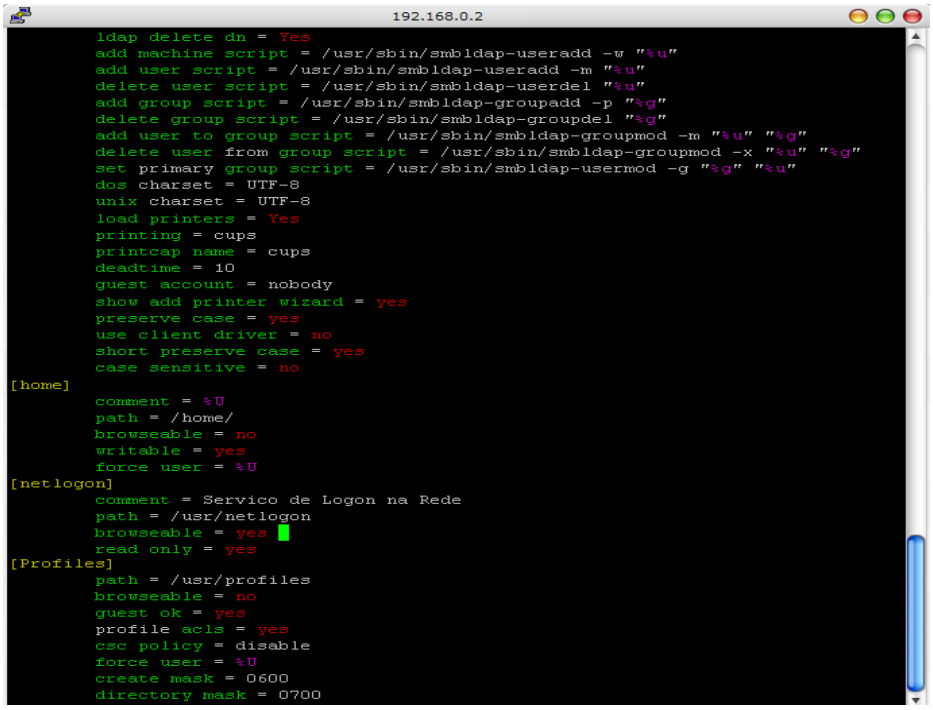

Figura 5.1.2: Segunda parte do smb.conf.

Na figura 5.2 mostra as configurações do home, dos usuário, netlogon, onde direciona os scripts de logon e o profiles, que define sobre regras dos perfis do controlador de domínio.

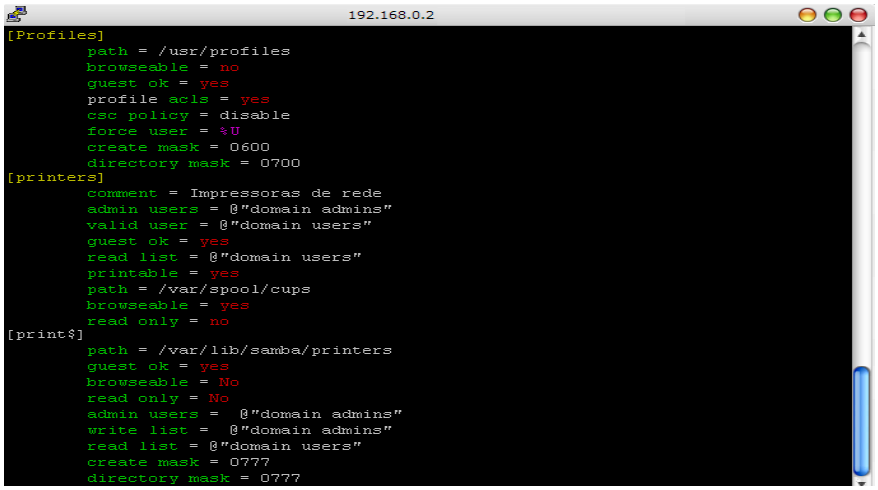

Figura 5.1.3: Terceira parte do smb.conf.

Na figura acima mostra a configuração sobre as impressoras pertencentes ao domínio sob controle do servidor PDC.

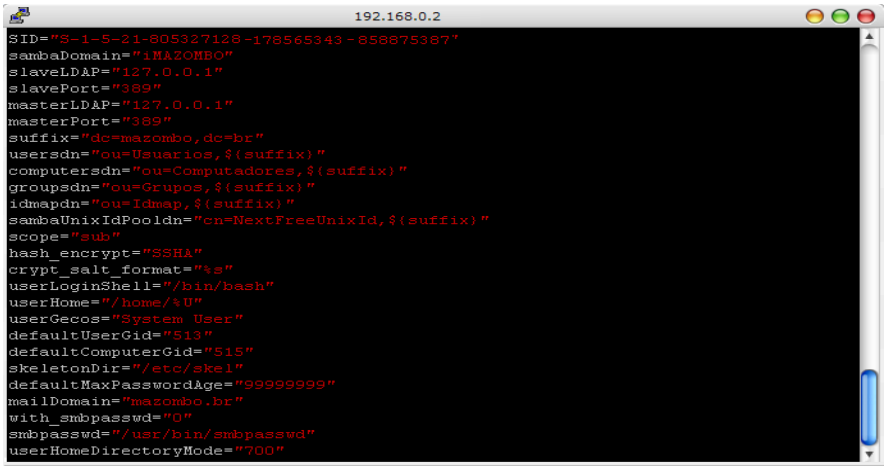

Figura 5.1.4: Arquivo smbldap.conf

A figura 5.4, mostra o arquivo de configuração smbldap.conf, onde consta alguns parâmetros necessários para a integração do samba com o ldap.

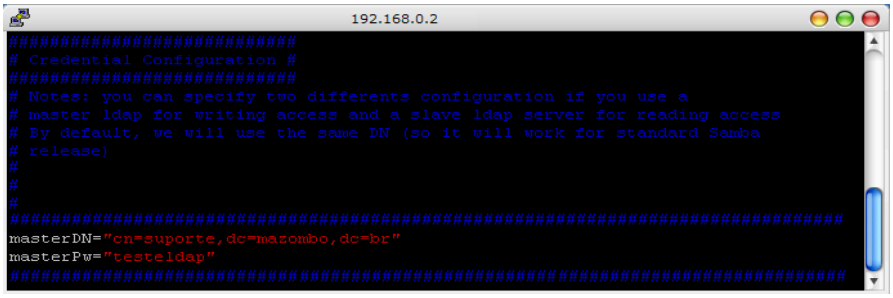

Figura 5.1.5: Arquivo smbldap-bind.conf

A figura acima mostra o arquivo onde armazena os parâmetros necessários para se acessar a base ldap, principalmente quando se usa alguma ferramenta ou software que existe suporte ao ldap.

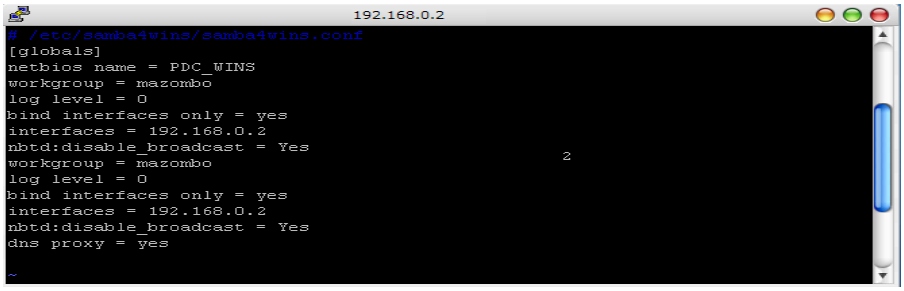

Figura 5.1.6: Arquivo samba4wins.conf

A figura 5.1.6 mostra o arquivo samba4wins.conf, onde se estipula parâmetros necessários para a replicação do serviço wins no domínio do PDC.

5.2 CONFIGURAÇÃO DO LDAP

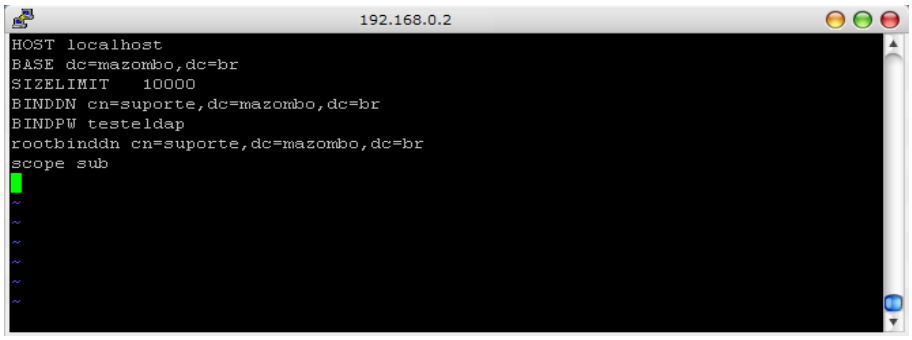

Figura 5.2.1: Arquivo ldap.conf

O arquivo ldap.conf, armazena configurações sobre a base do diretório ldap, como o nome do host, domínio da base e senha de acesso.

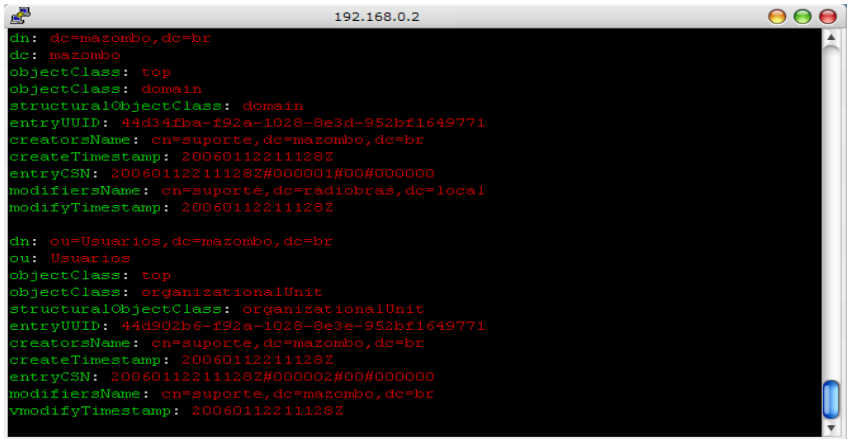

Figura 5.2.2: Primeira parte do arquivo mazombo.ldif

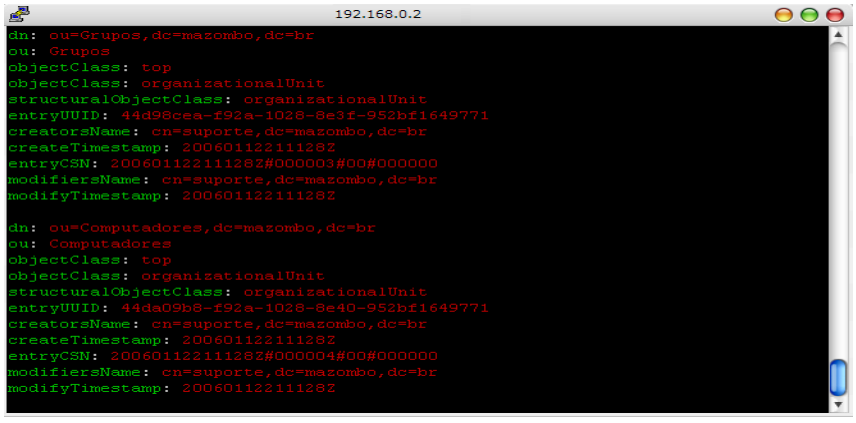

Figura 5.2.3: Segunda parte do arquivo mazombo.ldif

O arquivo mazombo.ldif armazena as informações sobre a base de dados do diretório (usuários, grupos, computadores, etc..) no servidor PDC do domínio mazombo.br.

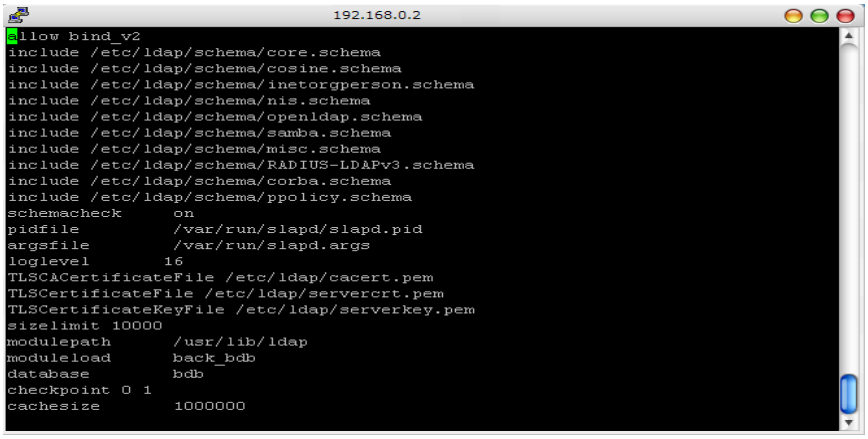

Figura 5.2.4: Primeira parte do arquivo slapd.conf

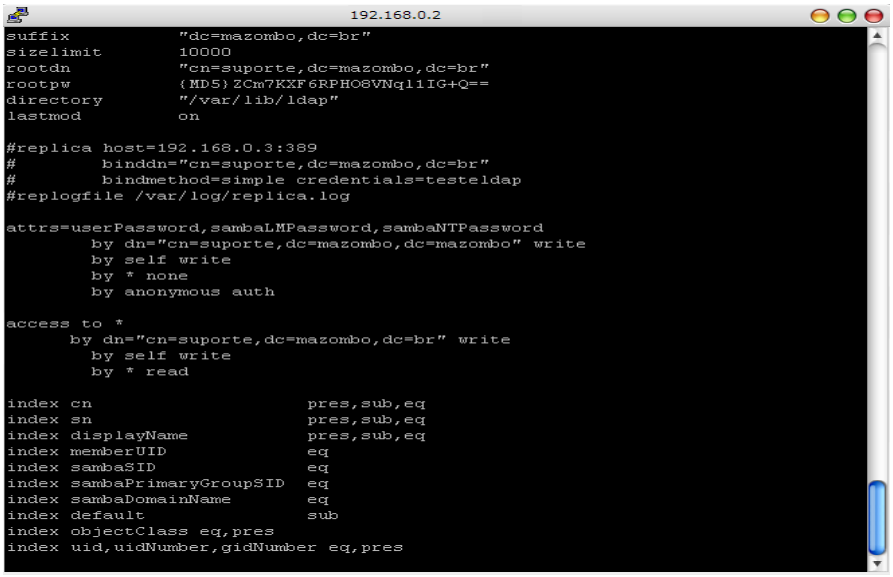

Figura 5.2.5: Segunda parte do arquivo slapd.conf

As principais configurações do LDAP estão armazenadas no arquivo slapd.conf, onde esta armazenado os parâmetros de todo o funcionamento do OpenLdap.

5.3 UTILIZAÇÃO DO SERVIDOR DE DIRETÓRIO PELO LDAPADMIN

Após o servidor configurado, é necessário utilizar uma ferramenta complementar para administrar a base de dados do LDAP. Por isso foi escolhido o LDAPADMIN, que possui fácil instalação e utilização, conforme demonstrado nas próximas figuras

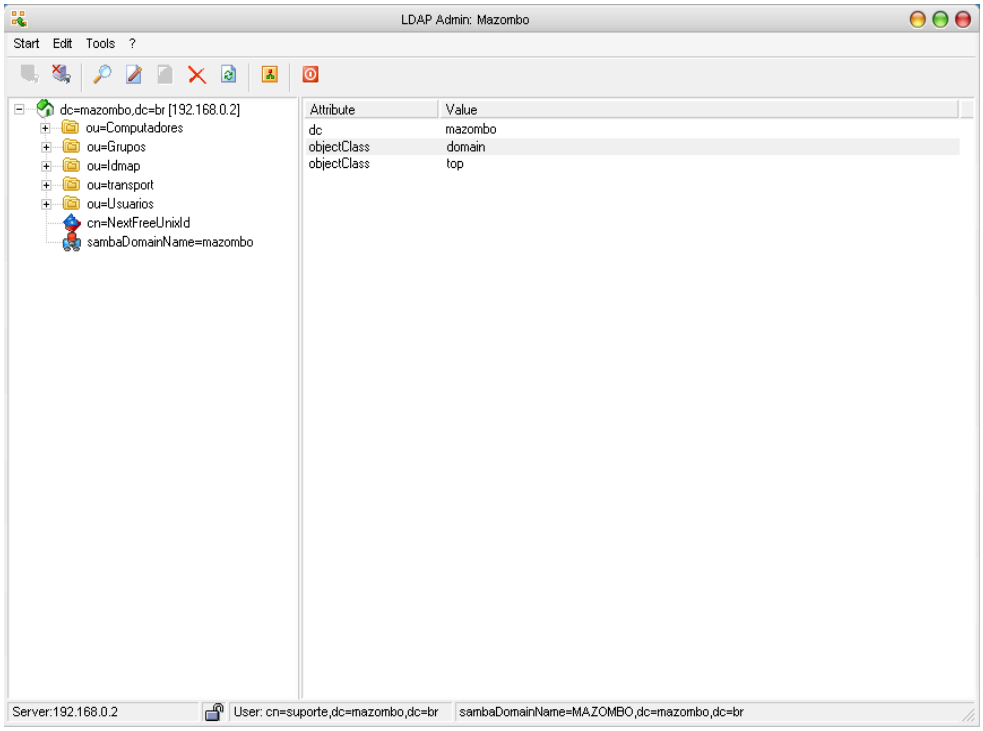

Figura 5.3.1: Tela inicial do LDAPAdmin.

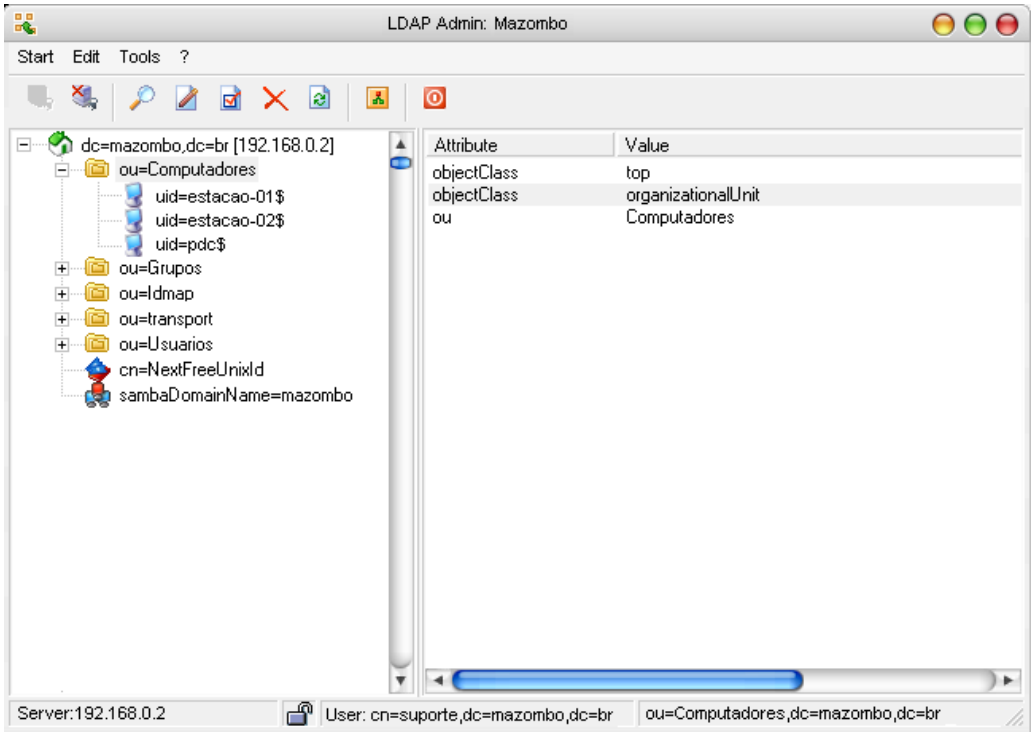

Figura 5.3.2: Acesso à base do diretório Computadores.

Na figura 5.3.2, mostra como é feito a consulta sobre os computadores que constam cadastrados na base de dados do LDAP.

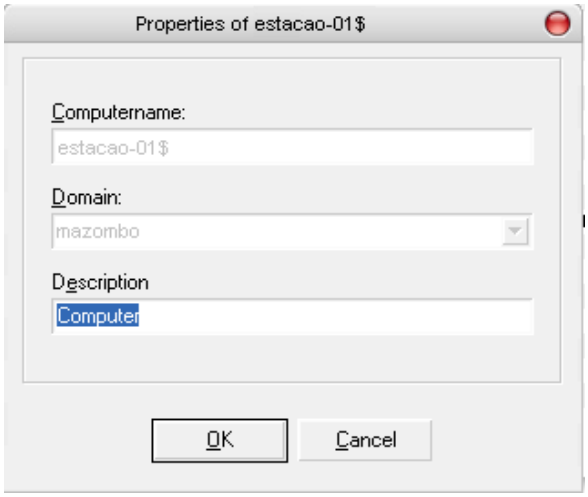

Figura 5.3.3: Cadastro de computador

O cadastro dos computadores na rede, na base de dados do LDAP é um procedimento simples como mostra a figura 5.3.1.

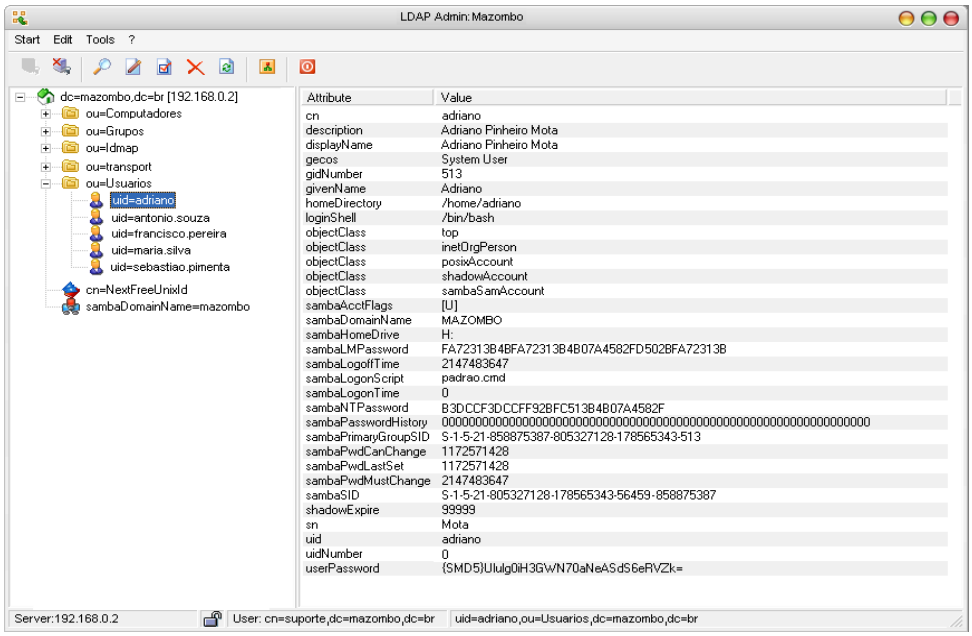

Figura 5.3.4: Acesso a base do diretório Usuário.

O acesso as informações dos usuários cadastrados na base do LDAP é demonstrado através da figura 5.3.4.

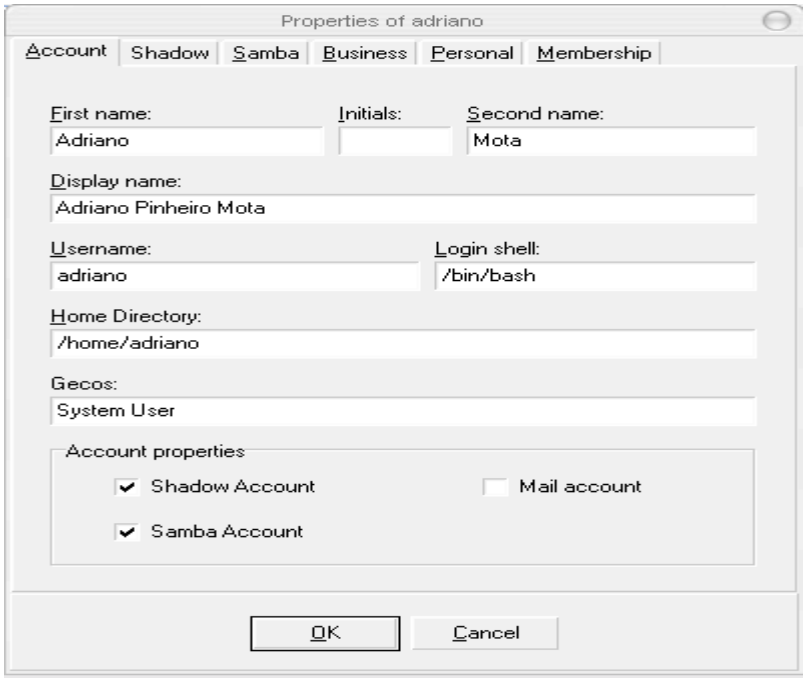

Figura 5.3.5: Primeira etapa para cadastrar um usuário.

Na figura 5.3.5 mostra a primeira etapa de cadastramento do usuário na base de dados do LDAP, onde encontra-se os dados referentes a conta na rede.

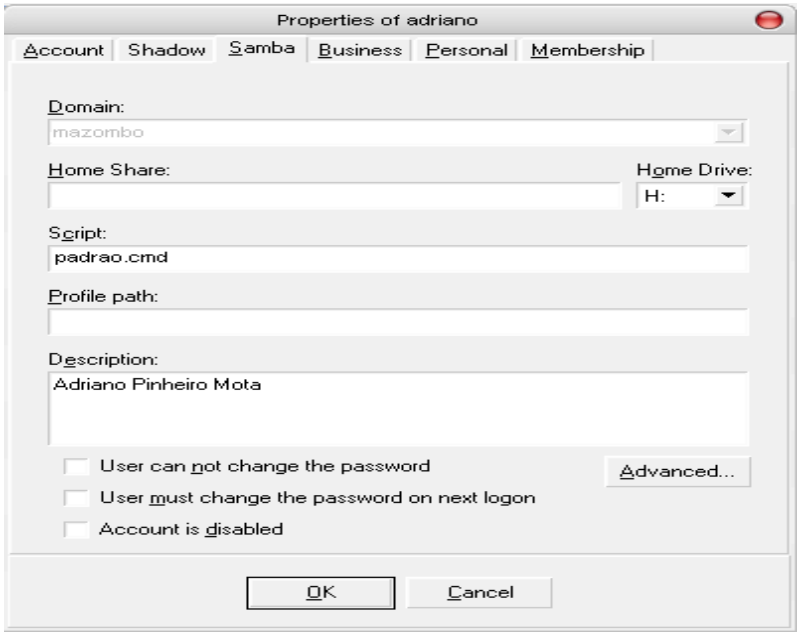

Figura 5.3.6: Segunda etapa para cadastrar um usuário.

A segunda etapa para cadastrar usuário consiste no preenchimento dos campos referentes às informações de acordo com o servidor Samba.

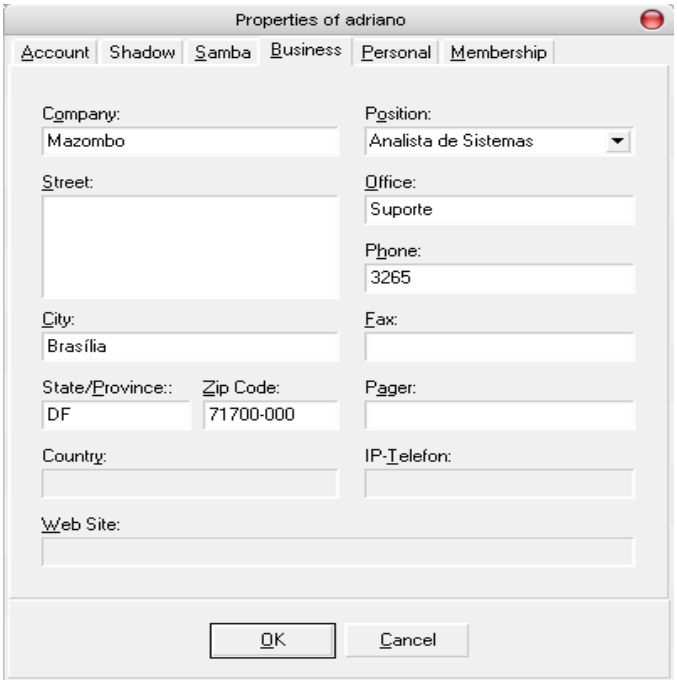

Figura 5.3.7: Terceira etapa para cadastrar um usuário.

Na terceira etapa do cadastramento de usuário é o local onde se colocam dados sobre o usuário, para identificá-lo se quando necessário.

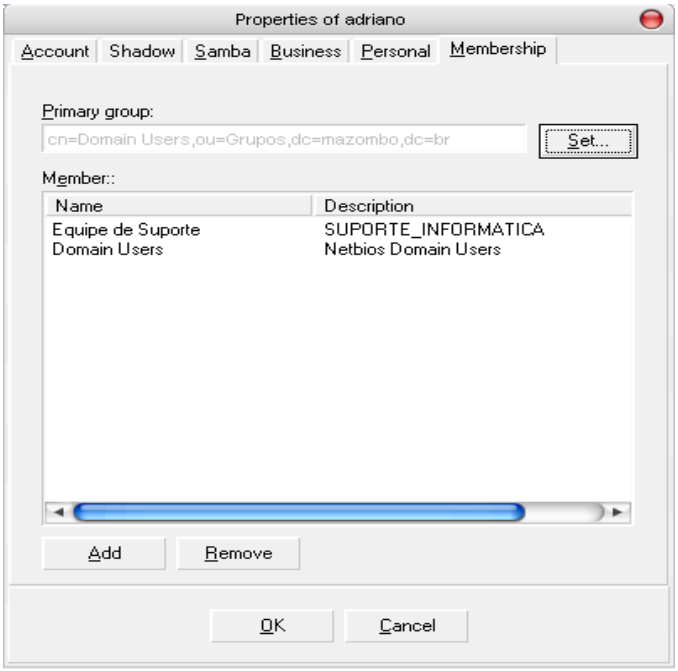

Figura 5.3.8: Quarta etapa para cadastrar um usuário.

Na figura 5.3.8 mostra o procedimento de cadastro de quais grupos o usuário irá pertencer.

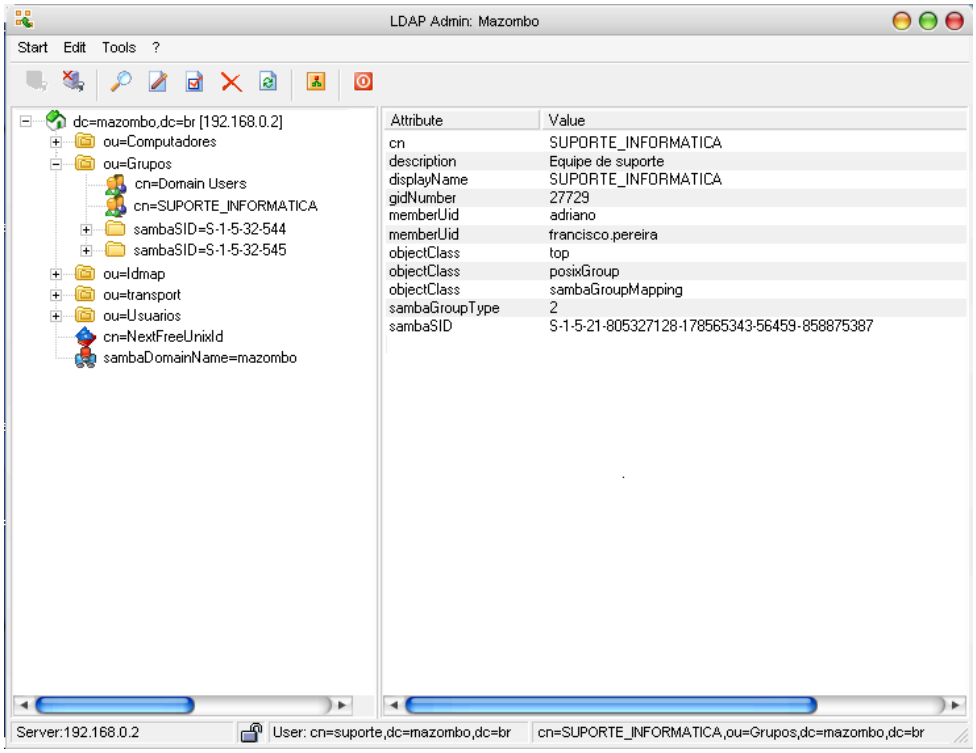

Figura 5.3.9: Acesso à base do diretório Grupos.

O acesso às informações dos grupos cadastrados na base do LDAP é demonstrado através da figura 5.3.5.

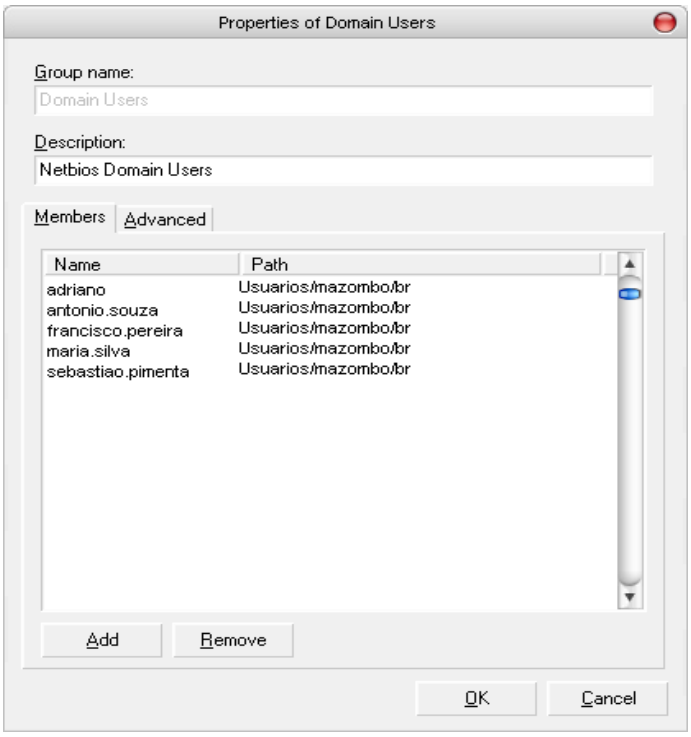

Figura 5.3.10: Primeira etapa de cadastro do grupo.

A figura 5.3.10 esboça o processo de cadastramento de dados de um determinado grupo.

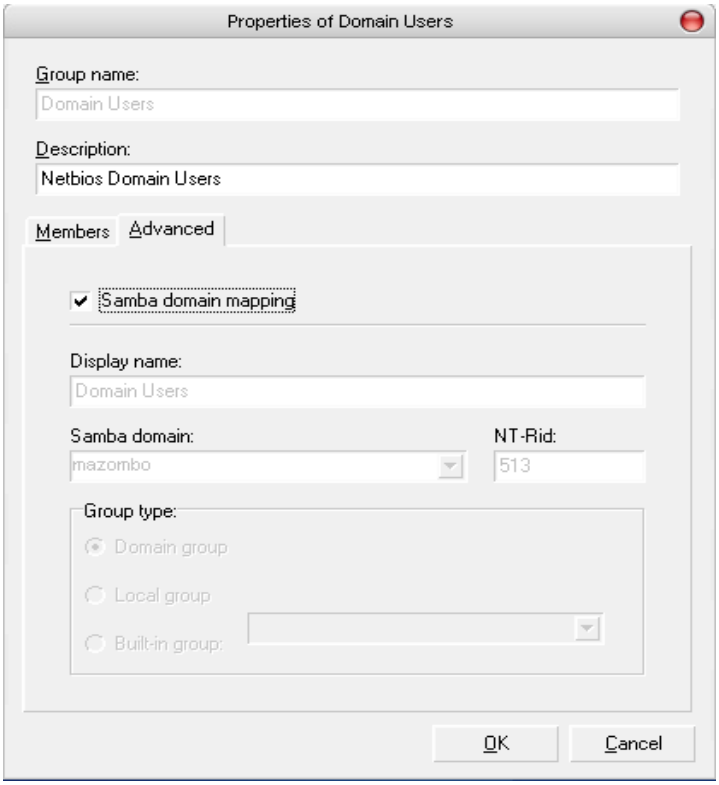

Figura 5.3.11: Segunda etapa de cadastro do grupo.

Na figura 5.3.11 demonstra a segunda etapa de cadastro de um determinado grupo, não há informações a serem cadastradas, pois a edição principal está na figura 5.3.10, na primeira etapa.

6. CONCLUSÃO

Atualmente na área de redes, muitos administradores, principalmente os que atuam para o poder público vem precisando se encaixar no processo de migração de serviços que se enquadrem no projeto do governo federal de software livre.

Para profissionais que atuam na iniciativa privada, levanta-se outras questões, como a questão de otimização de gastos e manutenção, onde muitas vezes é exigido uma solução que atenda as necessidades da mesma forma que as soluções proprietarias atenderiam.

Então foi focado a questão da utilização do PDC, um serviço extremamente importante para um funcionamento otimizado, organizado e operacionalmente viável. Avalio-se então a utilização do Samba integrado com servidor de diretório OpenLdap, uma solução extremamente funcional, mas com um detalhe, a questão de implementação e documentação, que em relação aos demais serviço é um procedimento pouco documentada e publicada pela comunidade.

Por isso realizou-se essa pesquisa para que pudesse reunir informações e experiências de sucesso na realização de tal procedimento e consequentemente poder dar uma contribuição para a comunidade.

Devido os dados coletados e expostos, constatou-se que a utilização do LDAP, oferece opções diversas para ser utilizado como solução e somando a ferramentas e recursos criadas que tem nativo conectividade com ele, a sua utilização é viavel para agilizar e dar qualidade ao trabalho de uma rede.

Com a realização dessa pesquisa sobre a integração do Samba com o LDAP, possibilitou um enorme acrescimo em relação aos conhecimentos e consequentemente um crescimento profissional. Então somados esses atributos, conclui-se que essa pesquisa foi de suma importância e sem ela talvez fosse inviavél a aquisição de detalhes existentes na solução em questão.

7. REFERÊNCIA BIBLIOGRÁFICA

BONAN, Adilson Rodrigues; Configurando e Usando o Sistema Operacional Linux. São Paulo: Ed. Berkeley, 2002. p.03-05, p. 373-398.

MORIMOTO, Carlos E.; Redes e Servidores Linux 2°Edição. Porto Alegre: Ed. Sul Editores, 2006. p. 216-241.

RIBEIRO, Uirá; Certificação Linux. Rio de Janeiro: Ed. Axcel Books, 2004. p. 181, p. 295-297.

HATCH, Brian; LEE, James; KURTZ, George; Segurança contra Hackers Linux 2°Edição. São Paulo: Ed. Futura, 2003. p. 416-418, p. 427-428.

KANIES, Luke A.; Uma Introdução ao Ldap. Disponível na Internet via www. url: http: //br.geocities.com/cesarakg/IntroLDAP-ptBR.html. Arquivo capturado em 05 de fevereiro de 2006.

LDAP. Administrator Guide. Disponível na Internet via www. url: http://www.openldap.org/doc/admin23/. Arquivo capturado em 13 de janeiro de 2007.

SAMBA. Docs and Books. Dispon[ivel na internet via www. url: http://us4.samba.org/samba/docs/. Arquivo capturado em 14 de janeiro de 2007.

WIKIPEDIA; Lightweight Directory Acess Protocol. Disponível na Internet via www. url: http://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol. Arquivo capturado em 16 de fevereiro de 2007.

ELSON, David; Autenticação Linux Usando OpenLdap. Disponível na Internet via www. url: http://online.securityfocus.com/infocus/1427. Arquivo capturado em 12 de janeiro de 2006.

AMERSDORFER, Markus; Using OpenLDAP on Debian Woody to serve Linux and Samba users. url: http://home.subnet.at/~max/ldap/index.php. Arquivo capturado em 30 de agosto de 2006.

LEMAIRE, Jérôme Tournier Olivier. Smbldap-howto. Disponível na Internet via www. url: http://www.idealx.org/prj/samba/smbldap-howto.en.html#htoc1. Arquivo capturado em 30 de agosto de 2006.