Directory Services with OpenLDAP: A Production Environment's Migration Case

João Paulo de Lima Barbosa e Olavo José Luiz Junior

UNIPAN – União Pan-Americana de Ensino, Paraná, Brasil

{joao,olavo}@unipan.br

Alessandro Kraemer

Faculdade de Pimenta Bueno, Rondônia, Brasil

si@fap-pb.com.br

Giovani Motter

PUCPR – Pontifícia Universidade Católica do Paraná, Paraná, Brasil

gmotter@ppgia.pucpr.br

Abstract. Actually, free software migration and deployment are topics widely discussed among different researches around the world. This paper presents a successful case of a free software system migration using OpenLDAP, and enumerates the steps followed to perform that and the issues found during the process. This research also presents the involved softwares limitations and potentialities, the integration level reached for each service, and the migration details.

Key-Words: Free Software, Open Source, Directory Services, OpenLDAP, Authentication Services.

Serviço de Diretórios com OpenLDAP: Um Caso de Migração em Ambiente de Produção

Resumo. Atualmente, a migração e implantação de sistemas de software livre são temas muito discutidos entre pesquisadores no mundo todo. Este artigo propõe-se a apresentar um caso de sucesso de migração de um ambiente de *software* livre utilizando OpenLDAP, demonstrando os passos utilizados para atingir este objetivo e também os problemas encontrados durante a migração. O trabalho também apresenta as limitações e potencialidades dos *softwares* envolvidos, o nível de integração alcançado em cada serviço, e os detalhes da migração.

Palavras-chave: Software Livre, Código Aberto, Serviços de Diretório, OpenLDAP, Serviços de Autenticação.

1. Introdução

A utilização de *software* livre nas empresas é uma realidade perceptível nos dias de hoje, principalmente quando a área em questão são serviços críticos em servidores.. Essa proximidade com o *software* livre e *open source* (código aberto) é ainda mais nítida quando se observa ambientes acadêmicos, ou seja, Universidades e Faculdades de forma geral (BRANDÃO *et al*, 2005, p. 54).

Independente do modelo de licenciamento dos *softwares*, sejam eles livres ou proprietários, em instituições de ensino ou ambiente corporativo como um todo, algumas questões de ordem de planejamento persistem a gerar problemas nas organizações. Uma das questões de maior impacto é a obsolescência de tecnologia, o que obriga a empresa a inovar inevitavelmente. A área de TI (Tecnologia da Informação) como um todo é extremamente mutável, e com uma velocidade assustadora surgem novos *software* e *hardware*, novas linguagens de programação, novos modelos de gerenciamento de projetos, etc. Essas inovações vêm sempre com o objetivo de melhorar a forma como algo é feito, e essas melhorias podem ser vistas sob várias perspectivas: melhor interface, maior desempenho, maior precisão, menor custo, ou seja, melhoria de forma global. Na área de TI essas inovações são vistas sempre com muito otimismo, porém adotá-las muitas vezes não é trivial, ou mesmo, factível. Considerando-se a implantação de uma tecnologia ou serviço em um ambiente que não esteja em produção, ou que este novo serviço não tenha dependências de dados não padronizados a serem importados de outros sistemas, espera-se que sua implantação ocorra sem grandes complicações, inclusive sobre o tempo de implantação. Porém se a realidade do ambiente for outra, com migrações de bases de dados a serem feitas, serviços e *hardware* substituídos, e que a migração ou implantação deva ocorrer com o ambiente em produção, os processos podem se tornar complexos. Isto ocorre devido ao *downtime*[1] indesejável dos servidores e, consequentemente, aos problemas operacionais causados aos usuários.

Outro problema é o planejamento de crescimento, pois envolve delimitação do ambiente de TI, perspectiva de crescimento da demanda por serviços e tecnologias, definição de tecnologias equipamentos e *software* de serviços a serem utilizados, que permitam esta escalabilidade. Segundo Audy e*t al* (1999, p.2), “Em virtude desta importância e do elevado investimento necessário para incorporar as novas tecnologias, as organizações devem procurar um máximo de garantias para viabilizar seu uso com sucesso”.

O crescimento tende a não ser um problema, quando este é bem planejado. Todas as condições anteriores conduzem à inovações, migrações, alteração de tecnologias, ou seja, mudança no ambiente de TI, e isso, como já mencionado, é inevitável. A diferença está no planejamento anterior, de forma a minimizar impactos, facilitando alterações quando necessárias.

O objetivo deste artigo é demonstrar os principais problemas encontrados na migração de um ambiente de TI de porte médio, onde o foco é a integração dos métodos de autenticação de serviços como Proxy e Samba, entre outros, através de OpenLDAP. Serão abordados os pontos mais relevantes da migração de cada serviço, bem como a ordem de migração utilizada nos mesmos e os respectivos motivos, especificando cada *software* e suas limitações, individuais ou agregados a outros serviços.

---

[1] Percentagem de tempo em que um sistema de computador, ou um de seus componentes, permanece inativo por causa de um problema inesperado ou para fins de manutenção, troca de equipamento, arquivamento de dados antigos, etc.

2. Metodologia

A migração será abordada sob a perspectiva da utilização de *software* livre, seja como serviço ou como ferramenta de apoio a migração. Basicamente se trata de conversão de dados entre padrões distintos, buscando enfatizar as limitações e possibilidades tanto dos *softwares* antigos como dos novos.

A base de dados do serviço Samba, por exemplo, era mantida em arquivos texto próprios do samba. Com a migração, o desejado é que essa base esteja representada em OpenLDAP, o que facilita a integração com outros serviços.

O processo de migração considera as seguintes fases:

1ª - Conhecimento do ambiente original – será detalhado o ambiente original, apontando suas características e forma de funcionamento, que trata a sessão 3.

2ª - Conhecimento da tecnologia de integração – será explanado a tecnologia de integração a ser utilizada, o OpenLDAP, apresentando um breve histórico sobre o mesmo, que trata a sessão 4.

3ª – Migração dos serviços mantendo o ambiente operável – nesta etapa será apresentado os detalhes da migração, apresentando cada software e serviço, e ainda apresentado as principais dificuldades encontradas na migração de cada serviço, que trata a sessão 5..

A figura 1 a seguir, demonstra em forma de diagrama o ambiente original de autenticações, as setas indicam para onde convergem as buscas por credencias de acesso no momento de uma autenticação. É possível observar que o gargalo de requisições está sobre o servidor Samba/NIS Acadêmico, entretanto observa-se três locais contendo bases de usuários: Samba/NIS Acadêmico, Samba Administrativo e Jabber.

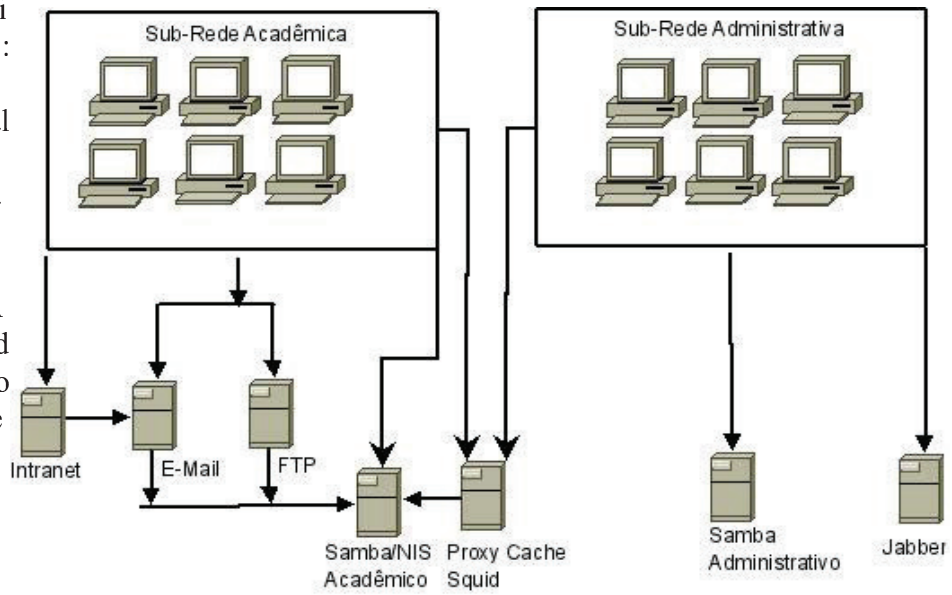

Figura 1: Ambiente original de autenticações

A presentado o ambiente original, é possível observar na figura 2 o ambiente proposto, que mesmo tendo aumentado o número de servidores envolvidos, tem-se uma clareza maior sobre o processo de autenticação. A ampliação do número de servidores objetiva o fracionamento de serviços, como é o caso do squid, e ainda a incorporação de novos serviços como é o caso do *captive portal*. Observa-se, também, uma base única de usuários, o OpenLDAP.

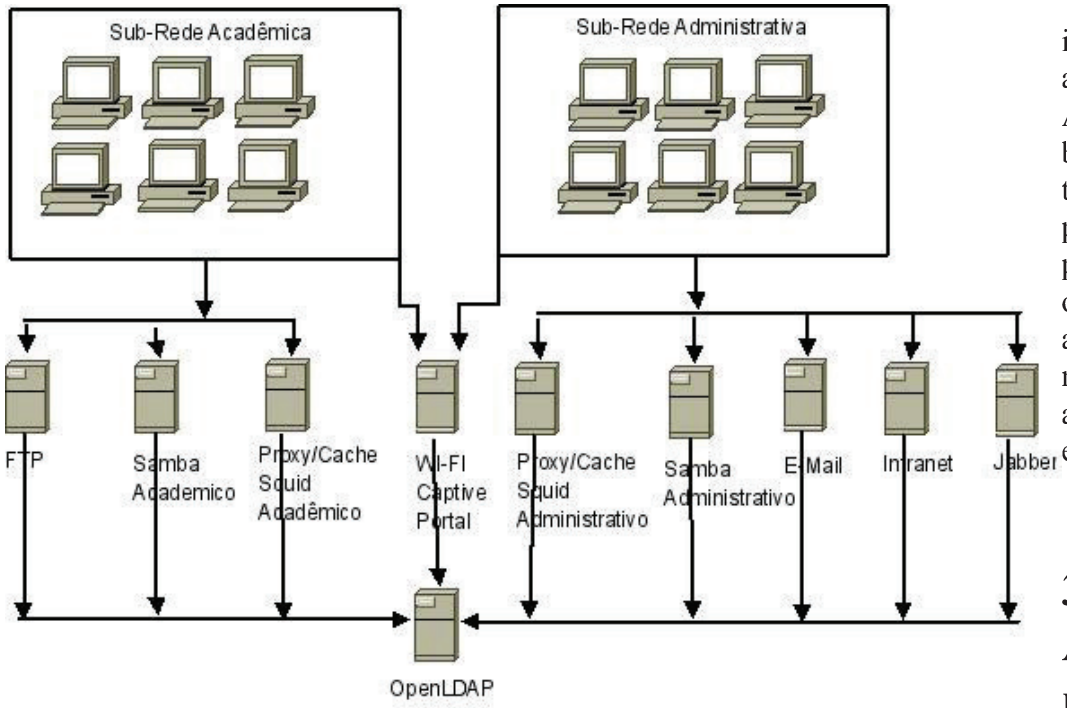

Figura 2: Ambiente proposto de autenticações

3. Ambiente Original

O ambiente original ao qual este artigo se baseia para o caso de migração pode ser descrito como parcialmente integrado, porém bastante limitado quanto à incorporação de novas aplicações que demandassem dados comuns. A migração ocorre de software livre para software livre. A instituição explorada cresceu de forma exponencial, porém, pouco planejado em alguns aspectos, o que levou a uma necessidade de inovações em diversos serviços. Não se trata de um cenário de missão crítica, no entanto, praticamente todas as inovações passavam necessariamente por migração de dados com o ambiente em produção. O ambiente computacional em questão trata-se de uma instituição de ensino de médio porte, com mais de 4000 acadêmicos e aproximadamente 250 funcionários e 250 estações de trabalho entre laboratórios e departamentos administrativos, porém em constante crescimento, o que demandava uma rápida integração entre as aplicações utilizadas por todos os envolvidos. Aplicações estas, comuns em ambientes corporativos de médio e grande porte, tais como autenticações de *e-mail*, ftp, *desktop*, *jabber*, intranet corporativa, e outras informações relativas a controladores de domínio e cadastro básico de usuários, contendo nome completo, CPF, telefone, RA de aluno, etc. Certamente cada aplicação demandou um estudo profundo e especifico. As subseções a seguir descrevem os serviços envolvidos nesta migração.

3.1 *Samba*

O primeiro serviço a ser "conectado" à base LDAP é o Samba. Portanto, vale a pena explorá-lo primeiro. Até então existiam dois servidores PDC[2]: um acadêmico e outro administrativo. Estes PDCs fornecem autenticação e compartilhamento de arquivos via

[2] PDC – *Primary Domain Controler* – Controlador de Domínio Primário – Trata-se de um serviço original do MS Windows NT,atuando de forma a centralizar informações e dados de um grupo de trabalho, mantendo este grupo de forma restrita. Este recurso é plenamente implementado pelo *Samba*.

samba a estações MS Windows, cada um fornecendo os serviços a que são destinados as suas respectivas sub-redes, acadêmica e administrativa.

Os usuários samba eram armazenados nos clássicos arquivos *smbpasswd*, contendo dados da conta do usuário (NT e LM *Password*, *username*, uid unix e flags) e dados de contas de máquinas, centralizados em um único arquivo. As informações do domínio PDC eram mantidas no diretório */var/lib/samba* em vários arquivos .tdb.

A divisão de sub-redes para o samba não era algo exato, pois o servidor samba acadêmico armazenava também em duplicidade as contas dos usuários administrativos - necessário para os serviços de e-mail e *proxy* cache *Squid*. Isto se deve porque esses serviços realizam suas autenticações de forma vinculada ao servidor PDC acadêmico, seja ele via samba ou via NIS.

3.2 NIS + NFS

As mesmas máquinas que atuam como servidores PDC exportam também seus usuários, grupos e senhas via NIS[3] para os serviços de *e-mail*, ftp e estações Linux para autenticação. Os compartilhamentos de diretório *home* de acadêmicos e professores são exportados via NFS para FTP, Servidor *WEB* e estações de trabalho Linux para atuarem em conjunto com o NIS, fornecendo autenticações e arquivos aos usuários.

3.3 Servidores de *E-Mail*

Os servidores de *e-mail* (POP, IMAP, SMTP) formam uma das partes mais complexas do processo de migração. Postfix é o serviço SMTP executado. Complementarmente, Dovecot é o serviço que trata dos protocolos IMAP e POP. Para o ambiente *webmail* é utilizado o squirrelmail.

O principal problema é o modelo de armazenamento de mensagens utilizado nesses serviços *Mailbox*, ou seja, cada caixa consiste em um único arquivo. No ambiente explorado há casos de caixas de *e-mail* com mais de 1,5 GB, criando arquivos com mais de 1,5 GB e fazendo com que o servidor manipulasse esses arquivos a cada *e-mail* recebido, lido ou apagado pelo usuário, o que estava onerando de forma crítica o servidor de correio eletrônico.

É importante reforçar que cada usuário da rede obtem um *e-mail*, sendo que são mais de 4000 contas e a maioria delas (65% aproximadamente) não eram utilizadas, ficando apenas recebendo lixo eletrônico (*spam*), o que aumenta o espaço utilizado em disco e aumenta o tamanho dos *backups*.

Outro ponto problemático é o formato de cota utilizada para o *e*-mail, a qual era utilizada através do sistema de arquivos (*file system ext3*). Devido à sua forma de funcionamento, quando uma caixa excede seu tamanho, o usuário não consegue acessar a caixa postal, fazendo-se necessária intervenção de um administrador para aumentar o espaço da caixa e posteriormente nova intervenção do administrador para a redução da cota assim que o usuário removesse as mensagens excedentes. Tudo isso ocorre devido ao

[3] NIS - *Network Information Service* – Trata-se de um serviço desenvolvido pela Sun Microsystem, para distribuição de informações em um rede, Com objetivo de manter uma base centralizada. A base de dados NIS é criada a partir de tabelas (plain text database), tal como /etc/passwd, /etc/shadow e /etc/group. O NIS também pode ser utilizado para outras tarefas mais especializadas (como para /etc/hosts ou /etc/services). O NIS efetua sua comunicação através de chamadas RPC(*Remote Procedure Call*).

formato de armazenamento *mailbox*.

A dependência do serviço cliente NIS no servidor de correio, tanto para autenticações dos usuários quanto para o controle de cota é outro problema destacado.

Por fim, outra limitação é a criação de *alias* de *e-mail*, que fica restrito ao arquivo do sistema */etc/aliases*. O armazenamento de contatos do *webmail* também é complexo, pois os contatos ficam em arquivos da pasta */var/lib/squirrelmail/data/*, o que dificulta o gerenciamento por ferramentas externas e ainda não havia nenhuma forma de agrupar todos os contatos de *e-mails* de funcionários e professores de forma automática e em todos os catálogos dos usuários. Assim, não é possível a exclusão automática de um contato/usuário da lista de *e-mail* quando o mesmo fosse desligado da instituição. O servidor de correio eletrônico é de longe o que mais recebe alterações visíveis para o usuário final.

3.4 *Squid*

A autenticação do *squid* ficava vinculada ao *samba* através o utilitário *smb_auth* do *squid*, autenticando-se no servidor *samba* acadêmico, portanto uma parada no serviço do *samba* acadêmico significava também uma parada no servidor *proxy cache squid*, pois o mesmo dependia do *samba*, haja vista que era utilizado o método clássico de autenticação do *squid* via *samba*. Este é um dos modelos de autenticação do *squid* mais utilizado em todo o mundo, porém não mais atendia as necessidades da instituição.

3.5 *Jabber*

O serviço de mensagens instantâneas utilizado pelos funcionários da instituição é implementado através do software *OpenFire*, e a base de usuários era mantida de forma paralela em banco de dados *MySQL*, não integrada com o restante das aplicações, gerando uma grave redundância de informações.

3.6 FTP

O serviço FTP provido através do *vsFTP* funciona satisfatoriamente, porém, depende do serviço NIS, o que faz com que o sistema que hospeda o serviço necessite conhecer todos os usuários, grupos e senhas. Ou ainda, todos os usuários do serviço FTP também precisam ser conhecidos pelo sistema operacional, e não apenas pela aplicação.

3.7 Intranet e Sistemas Legados Autenticados

As aplicações da Intranet corporativa, com seus diversos módulos, necessitam naturalmente de controle de acesso, ou seja, autenticação. No ambiente implementado na instituição a autenticação é feita de uma forma pouco convencional, onde se utiliza o servidor de *e-mail* para realizar essa autenticação.

Com a Intranet programada com *script s*em PHP, o processo de autenticação consiste basicamente em uma função em PHP que abre um *socket* e tenta se conectar e autenticar no servidor de *e-mail*. Se a autenticação no servidor de correio fosse bem sucedida, era buscado em uma base *MySQL* ou arquivo, o nível de acesso do usuários autenticado em questão, colocando-o assim em uma escala de privilégios de acesso.

3.8 *Wi-Fi*

O controle de acesso à rede sem fio também pode ser considerado complexo, já que a princípio apenas professores e funcionários devem ter acesso. A dificuldade de controle se dá pela necessidade de configurações de forma manual nos clientes *wireless*, no servidor *gateway e* no *access point*. Este cenário se tornou inviável quando surgiu a necessidade de liberar o acesso *wi-fi* controlado para os alunos.

3.9 Dependências Entre Serviços

A fim de levantar dados importantes sobre a migração foi construída a tabela a seguir que consolida os principais serviços envolvidos, bem como suas dependências e grau de integração. Através desta tabela é possível observar que o Samba não possui dependências e por isto deve ser adaptado (migrado) primeiramente.

Serviço	**Serviços Dependentes**
Samba Administrativo	Nenhum
Samba Acadêmico	Nenhum
Squid	Servidor Acadêmico/*Samba*
Servidor de *E-Mail*	Servidor Acadêmico/NIS
FTP	Servidor de Acadêmico/NIS, Servidor Acadêmico/NFS
Intranet e Autenticações	Servidor de *E-Mail*
Jabber (*OpenFire*)	Nenhum
Wi-fi	Roteador/*Squid*
Estações Windows	Servidor Acadêmico e Administrativo/*Samba*
Estações Linux	Servidor Acadêmico/NIS/NFS

Tabela 1: Relação de dependência entre serviços nos ambiente original

4. Definição da Tecnologia para Integração

A instituição explorada no contexto deste artigo sempre obteve uma relação muito estreita com o *Software* Livre. Nesta instituição, só não é utilizado software livre quando não existe alternativa equivalente às necessidades. Isso se deve à proximidade que os gestores de TI da instituição mantêm com essa classificação de *software* devido à sua filosofia, por fazer todo o sentido esta filosofia colaborativa, bem como por questões de robustez e performance. Outro aspecto positivo, é a redução de despesas com licenças. Tal proximidade criou na instituição um ambiente misto, de forma que *softwares* proprietários

são utilizados paralelamente com *softwares* livres. Por exemplo, em estações de trabalho executando Microsoft Windows e *Softwares* Livres como o GNU/Linux Debian, Slackware, Suse e CentOS.

Na instituição explorada no contexto desse artigo, também no que tange às aplicações, as proprietárias convivem em paralelo com aplicações de software livre, onde por um lado temos servidores de correio, web, ftp, dns, chat, arquivos, vpn entre outros rodando software livre, por outro temos sistemas ERP e de gestão acadêmica sendo softwares proprietários e rodando sobre plataformas proprietárias e livres.

Os recursos financeiros para TI são limitados na instituição explorada, e uma vez que o departamento acaba de receber um grande investimento em infra-estrutura (Servidores, Switchs, Storages, etc.), gastos com licenças e *softwares* proprietários era algo incoerente com a filosofia de trabalho hora implantada, uma vez que a equipe de TI da instituição tem como princípio a utilização de *softwares* livres.

O objetivo central, que é a integração dos diversos serviços oferecidos pela instituição visando atender acadêmicos e funcionários, a etapa inicial é a definição de uma forma de armazenamento centralizado de informação, principalmente autenticação, cujo foco é seu compartilhamento entre diversas aplicações.

Para realizar a centralização de informações é necessária a implementação do protocolo LDAP. Em se tratando de software livre, isto pode ser feito através do OpenLDAP. Um fator importante para essa tomada de decisão é a divulgação de casos de sucesso, principalmente no segmento publico federal e estadual, como são apresentados os trabalhos de Chrispiniano (2005) e Governo Federal (2004). Também deve ser considerado que os serviços em migração devem possuir suporte a LDAP.

4.1 LDAP

De forma a evitar redundâncias e retrabalho, comumente se pensa em centralização de informações. Porém centralizar informações e ainda permitir que diferentes aplicações tenham acesso a essa informação não é tarefa fácil, pois implica na padronização das tecnologias, o que leva a encontrar um protocolo adequado – o LDAP. Desta forma, fabricantes de hardware e desenvolvedores de software podem criar aplicações compatíveis entre si.

A ITU Telecommunication desenvolveu as especificações X.500 a qual faz parte o protocolo de acesso a diretórios DAP, que vinha de encontro com as necessidades de centralização das informações (Yeong, 1993). Porém, segundo Trigo (2007), este protocolo foi desenvolvido baseado no modelo de referência OSI (*Open System Interconnection*). O OSI é tido como um modelo de transmissão de dados pré-Internet, extremamente difícil de ser implementado corretamente, entre outros problemas, o que acabava por resultar em aplicações pesadas e lentas.

Com o surgimento da Internet o protocolo TCP/IP foi mais bem aceito, e esse problema foi resolvido com a criação de um protocolo que se encaixasse melhor nos moldes TCP/IP. Estava então definido o novo modelo de integração e centralização de dados, o LDAP (*Lightweight Directory Access Protocol*) - Protocolo Leve de Acesso a Diretórios, padronizado em 1993, no RFC 1487 da IETF (*Internet Engineering Task Force* – Força Tarefa de Engenharia da Internet). O LDAP se encontra na sua terceira versão, conhecida como LDAPv3, especificado em uma série de RFC, como apresenta o RFC 4510.

Turtle (2003) apud (TRIGO, 2007, p.22) afirma “(...) o LDAP é um padrão aberto

capaz de facilitar, de forma flexível, o compartilhamento e a manutenção de grandes volumes de informações, definindo um método padrão de acesso e atualização de informações dentro de um diretório."

LDAP não armazena nenhuma informação, ele é somente responsável por viabilizar o acesso as informações (ou protocolo de acesso as informações). As informações podem estar armazenas desde em arquivos texto até em bases de dados relacionais, esta por sua vez é denominada *backend* (MACHADO; MORI, 2006). A base de dados utilizada neste caso de migração é o BDB4 (Berkley Data Base 4),. O BDB4 é a base instalada por padrão juntamente com o OpenLDAP. Trigo (2007, p. 195) da exemplos da utilização do LDAP com *backend* MySQL, que é outra alternativa viável.

Ao logo do tempo, várias empresas desenvolveram suas implementações do protocolo LDAP, algumas das mais conhecidas são o *Active Directory* da Microsoft Corporation, o NDS da Novell, o *iPlanet* da SUN e o *Directory Server* da Netscape (Howes, 2003). Todos são serviços de Diretórios satisfatórios, e mesmo sendo baseados em padrões abertos, tem o inconveniente de ser implementação proprietária, o que gera custos e algumas limitações. Buscando atender a comunidade do Software Livre surge o OpenLDAP.

4.2 OpenLDAP

OpenLDAP se apresenta como uma solução Livre de auto nível e performance, é um software livre licenciado pela GPL (*General Public License*– Licença Publica Geral) e desenvolvido inicialmente pela Universidade de Michigan, cumprindo de forma plena toda a especificação do protocolo LDAP.

Com a necessidade cada vez maior de centralização de informação, aliado ao formato de licenciamento, o OpenLDAP ganhou espaço nas grandes corporações, principalmente nos segmentos públicos brasileiros, que com seus *casos* de sucesso acabam por influenciar empresas do segmento privado a adotar essa solução.

O OpenLDAP é utilizado para propiciar o acesso ao serviço de diretórios através do protocolo LDAP para vários tipos de aplicações, porém, as que mais se destacam são armazenamento de dados de usuários para autenticação (ABADE, 2004), como senha, nome, telefone, diretório *home*, endereço, informações de acesso, última troca de senha, e qualquer informação que deva ser compartilhada entre diversos sistemas. Entretanto, é preferível que essas informações mantenham a característica de muita leitura e pouca alteração (gravação). Segundo Abade (2004), isso coloca o LDAP na categoria de banco de dados, porém um banco que dispensa a complexidade de uma base relacional e otimizado para leitura.

O suporte aos mais variados tipos de aplicação é provido através de *schemas*, que nada mais são que arquivos que definem campos e tipos de dados a serem utilizados por aplicações. Em uma analogia com uma base de dados relacional clássica, os *schemas* são equivalentes a definição de uma tabela (CARTER, 2003). Quando uma aplicação é construída (programada), e seus desenvolvedores pretendem que ela tenha suporte a LDAP através do OpenLDAP, existe a opção de utilizar *schemas* padrões que são pré-instalados juntamente com o OpenLDAP. Também é possível elaborar um novo *schema* mais adequado à aplicação. Esta segunda opção normalmente é utilizada quando se necessita de campos além dos da autenticação, e um bom exemplo para isso é a aplicação *samba* que armazena no serviço de diretórios uma série de informações relacionadas a cada usuário, grupo, e ao seu domínio PDC.

Apesar de existirem vários benefícios de se implantar um ambiente com

informações centralizadas utilizando o protocolo LDAP, implementações práticas demonstram que no momento da integração eventuais problema surgirão, tanto do lado do LDAP como das diferentes aplicações. Pelo lado do LDAP, um problema clássico a ser resolvido é o método de criptografia (*hash*) utilizado para criptografar a senha *unix*[4] do usuário. Neste cenário, alguns métodos possíveis são: *plain*, *crypt*, *sha*, *ssha1*, *md5*, entre outros. O problema é que uma vez definido o método a ser utilizado, deve-se certificar-se que todas as aplicações que irá fazer uso desse atributo (campo) tenham suporte a este método criptográfico.

5. Migrando os Serviços e Aplicações

Para todas as migrações de serviços e servidores que foram realizadas, deve ser entendido que as mesmas foram efetuadas em ambiente de produção na maioria dos casos. Nos casos de paradas dos serviços, os mesmos voltaram ao funcionamento em menos de duas horas. Apesar de não ser um ambiente de missão crítica, paradas longas são inaceitáveis. Outro ponto a ser considerado, é que em paralelo à migração de serviços foi realizada a substituição de 80% dos servidores do *datacenter* da instituição.

5.1 OpenLDAP

O primeiro serviços a ser instalado é o LDAP. Sua instalação foi realizada em um sistema GNU/Linux Debian 4.0 (*Etch*), e sobre um *hardware* ício a aplicações de banco de dados, com discos de alto desempenho (SAS 15000 RPM) e controladora de disco com *cache*. O OpenLDAP seguiu uma instalação básica, mantendo nesse momento apenas seus *schemas* padrões e *hash* senhas SSHA.

5.2 *Samba* Servidor Administrativo

No ambiente institucional são dois servidores Samba que devem ser migrados para trabalhar de forma integrada com o LDAP, conhecidos como *samba* acadêmico e *samba* administrativo, ambos para autenticação de usuários. Entende-se que devido às dependências sistêmicas esses servidores *samba* devem ser os primeiros a serem migrados.

O *samba* administrativo possui menos serviços dependentes que o servidor *samba* acadêmico. O samba acadêmico possui relacionamentos com o *squid*, *e-mail*, ftp, intranet e autenticação de estações Linux e MS Windows. Boa parte desses serviços dependentes não faz parte do *samba* e sim do serviço NIS e NFS que são executados na mesma máquina. Mas isto não altera a tomada de decisão em iniciar a migração pelo servidor samba administrativo, que é responsável apenas pela autenticação de estações MS Windows, possuindo menos dependências.

O ambiente chamado de administrativo, conta com aproximadamente 80 estações de trabalho que dependem diretamente do *samba* para autenticação e compartilhamento de

[4] Senha Unix – corresponde ao atributo userPassword do *schema* core.schema do OpenLDAP, trata-se de um *schema* padrão, e fornece informação da senha do usuário para diversas aplicações, como por exemplo: proFTPd, Postfix, Courier, Autenticação Unix/Linux, OpenFire (*Jabber*), entre muitas outras aplicações.

arquivos. O servidor *samba* administrativo funciona a partir de outro *hardware* (servidor), o que possibilita uma migração em ambiente de produção.

Antes de realizar uma migração oficial foram realizadas migrações em ambiente de teste com a finalidade de avaliar antecipadamente os impactos e possíveis problemas, contribuindo significativamente.

Como objetivo da migração não é desejável manter informações das contas de usuário no servidor *samba*, e sim que todas estas informações estivessem na base LDAP compartilhada com todos os serviços. O papel do samba é apenas gerenciar as contas de usuários.

A opção *smbldap-tools do servidor samba* consiste em um conjunto de *scripts* feitos em PERL que se encarregam do gerenciamento de contas *samba* na base LDAP, desta forma eliminando a necessidade de uma cópia local da conta. O *smbldap-tools* vem com um conjunto de *scripts* para facilitar a migração de usuários, senhas e grupos unix a partir de arquivos para a base *samba*/ldap. Porém, em nenhum momento ele se propõe a migrar os dados das contas *samba* encontrados no arquivo *smbpasswd*. O que ele faz é criar uma conta *unix* compatível com o *samba*, adicionando a esta o *objectClass*[5] *sambaSamAccount*[6], obrigatório em contas samba, no entanto a partir de seus *scripts* não é possível migrar a senha dos usuários, desta forma novas senhas do samba deveriam ser criadas para todos os usuários deste serviço, impactando na idéia de migração transparente para o usuário, o que não é desejável. Para tanto, um estudo mais aprofundado foi necessário, identificando cada elemento do arquivo, como usuário, *uid* unix, NT e LM *password* e flags.

Como solução para este problema de migração foi desenvolvido um *script* em *bash* que realiza a leitura do arquivo *smbpasswd* obtendo o usuário e senhas NT e LM para em seguida criar um LDIF[7] com essas informações e atualizar a base LDAP sem utilizar o *samba* ou o *smbldap-tools*. Certamente este processo se faz apenas uma vez, durante a migração de contas, sendo que a partir desse momento o *smbldap* passa a gerenciar as senhas NT e LM do *samba*.

Como já mencionado, o novo servidor *samba* administrativo entrou em produção de forma paralela com o antigo, onde este exportava os arquivos via NFS para o novo. Esse paralelismo permaneceu até que todas as estações fossem migradas para o novo domínio. Um novo nome de domínio foi criado para o novo servidor, permitindo com que ambos os servidores convivessem em paralelo por aproximadamente uma semana, ao fim da qual todas as estações administrativas estavam no novo domínio, com novo servidor *samba* e informações na base LDAP.

Este foi o primeiro passo operacional para uma série de alterações. A partir desse momento os serviços de caráter administrativo podem ser desvinculados do servidor samba acadêmico.

---

5 *ObjectClass* – Classe de Objeto – Utilizada para realizar a definição de uma classe de objeto (BUTCHER 2007).

6 *SambaSamAccount* – Classe de objeto pertencente ao *samba*, define uma entrada no LDAP como sendo pertencente também ao *samba*.

7 LDAP Data Interchange Format (LDIF) – é um padrão de texto plano para intercâmbio de dados formatado para representar os dados do LDAP, sendo obviamente um formato de registro para inserções e atualizações na base LDAP. (MACHADO; MORI, p. 58)

5.3 Squid Servidor Administrativo

O servidor Roteador e *Proxy Cache* também ganhou um novo contexto, onde inicialmente se tinha apenas um servidor *Proxy Cache* servindo as duas sub-redes (acadêmica e administrativa) e autenticando ambas no servidor acadêmico, no novo contexto trabalha-se com dois servidores roteadores *proxy cache*, um para a rede administrativa e outro para a rede acadêmica.

Tendo servidores *proxy* separados é possível realizar autenticações separadas, o que, da forma de autenticação utilizada anteriormente levaria a uma duplicação de bases. Complementarmente, a autenticação do *squid* deixou de ser através do *samba* e passou a ser diretamente na base LDAP, uma vez que o *squid* oferece suporte para o mesmo através de seu binário *ldap_auth*. Com isso, se desfez a dependência do *samba* para o funcionamento da autenticação do *squid*. Outro impacto observado é que desta maneira os servidores *samba* não são mais sobrecarregados para autenticação *squid*. A distribuição Linux utilizada nesta tarefa foi o Ubuntu Server 8.04 LTS. O uso do Ubuntu ao invés do Debian, que é a plataforma padrão para servidores Linux da instituição, se deve porque o Debian 4.0 (com seu *kernel* padrão 2.6.18) não reconhece algumas placas *Ethernet* nesses roteadores, e o Ubuntu, por contar com um *kernel* mais recente (2.6.24), reconheceu automaticamente essas placas.

5.4 *E-Mail*

Os problemas e limitações do serviço de correio fizeram-no se tornar um dos elementos mais complexos, de difícil administração dentro da rede. Como já mencionado anteriormente, o serviço de correio neste âmbito compreende smtp, pop3, imap, *webmail* e cliente de *e-mail* (MUA – *Mail User Agents*). Estes serviços eram oferecidos a todos os usuários da rede, sejam eles acadêmicos, professores ou funcionários. Entretanto, um alto percentual destas caixas de *e-mail* dos alunos não era utilizado pelos mesmos, o que gerava uma grande quantidade de caixas não utilizadas, porém ativas e recebendo lixo eletrônico, onerando o serviço de *backup*.

Após profunda análise, juntamente com a direção da instituição, ficou decidido descontinuar o fornecimento de contas de *e-mail* para os alunos, uma vez que a maioria não utilizava estas contas e alguns até desconheciam de sua existência. Desta forma, a instituição assumiu o compromisso de focar esforços e investimentos em recursos para os serviços que não são disponibilizados de forma prática e gratuita na Internet, uma vez que *e-mail* é oferecido por muitas empresas como Google e Microsoft de forma gratuita. Assim, não havia razões para continuidade de alocação de recursos para estes serviços. Estava então definido que em um novo cenário o correio eletrônico deixaria de existir para os alunos, mas se manteria para professores e funcionários.

Após tomada essa decisão surgiu a dúvida sobre como implementar o novo cenário, pois não é desejável que as contas sejam excluídas repentinamente, haja visto que mesmo sendo uma parcela pequena, alguns alunos mantinham este serviço como seu *e-mail* padrão. A solução adotada foi a publicação de edital no portal *web* da instituição anunciando antecipadamente sobre a descontinuidade do serviço e que em uma data prevista ele seria bloqueado para recebimentos e envios, podendo ser acessado apenas para consultas – também por um tempo determinado. Porém, ainda se mantinha uma dúvida: Como bloquear apenas as contas dos alunos? Pois o servidor de correio antigo ainda estava em produção e este valida seus usuários (contas de correio, no caso) no servidor *samba* PDC acadêmico através do NIS. Com isso, observou-se que era necessária alguma medida

de controle no próprio servidor de correio.

Após alguns contatos com a comunidade da área de informática e software livre, foi sugerida a ferramenta *apolicy*, ou ACL *Policy*, que é capaz de atuar juntamente com o *postfix* fazendo filtros de mensagens, gerando alertas, bloqueios, mensagens de resposta automática, redirecionamentos, entre outros, tratando-as através de ACL (*Access Control Lists*). Esse tratamento é muito similar a utilização de ACLs no *squid*. O *apolicy* define suas ações através dos alvos *postfix*, por exemplo, *ACCEPT*, *DROP*, *REJECT*, *REDIRECT*, entre outros. A instalação do *apolicy* para seu funcionamento juntamente com o *postfix se* deu de maneira satisfatória, criando uma lista de pessoas que poderiam enviar e receber *e-mails*, e, consequentemente, colocando nessa lista apenas funcionários e professores.

Desativado as contas de alunos, uma vez que já se encerrara o prazo para que os mesmos deixassem de utilizar as contas de *e-mail*, deu-se início a implantação do novo servidor de correio, que atende professores e funcionários disponibilizando os serviços de *smtp*, *pop3s*, *imaps*, *sasl*, antivírus, antispam, *greyliste webmail*com https.

Com a ajuda de Trigo (2007), iniciou-se a implantação do servidor. A distribuição GNU/Linux utilizada foi Debian 4.0 (*Etch*), que é a distribuição Linux padronizada para servidores da instituição, como mencionado anteriormente. Nesta fase do projeto se estudou o melhor modelo de armazenamento de mensagens e se optou por utilizar *maildir*. Também foi selecionado o melhor sistema de arquivos a ser utilizado para este propósito, optando pelo XFS, definindo inclusive qual seria o melhor tamanho de bloco para esta finalidade, sendo escolhidos blocos de 2K ao invés dos tradicionais 4K. A razão desta ultima escolha é buscar minimizar a fragmentação interna no bloco, visto que uma grande quantidade de mensagens de *e-mail* é composta unicamente por textos curtos, o que as deixam com 2K ou menos de tamanho em disco. Entretanto, como o tamanho mínimo de alocação é o bloco, blocos de 4K tendem a sofrer muita fragmentação com mensagens pequenas.

O objetivo é utilizar alguns campos na base LDAP que não são contemplados com os *schemas* padrões do OpenLDAP, foi utilizado um *schema* próprio para o serviço de correio. Curiosamente, mesmo utilizando o *postfix* como MTA, o *schema* utilizado foi o do *qmail*, que, segundo Trigo (2007), é utilizado pelo *postfix para*satisfazer todas as suas necessidades.

Para o serviço de SMTP foi utilizado o *postfix* compilado com suporte a cota virtual (através do *patch* VDA), *alias* (apelido) de *e-mail* e *forward* (encaminhamento) de *e-mail*, buscando todos estes na base LDAP. Aqui surge o primeiro detalhe a ser ajustado, o *postfix* na versão disponível (2.3.8) nos repositórios oficiais do Debian 4.0 não trabalha com o protocolo LDAP versão 3, somente com a versão 2. Para resolver este problema foi necessário incluir uma linha no arquivo de configuração do OpenLDAP, ”instruindo” o mesmo a rodar em compatibilidade com a versão 2. Do contrário, o *postfix* não se comunicaria com o LDAP.

Outro problema ocorreu ao longo da configuração do *postfix* no momento de fazer configuração para autenticação SASL. O *software* utilizado para efetivar a autenticação é o *courier-authdaemon*, o mesmo utilizado para *pop3s* e *imaps*, explorados mais adiante. O problema é que o *postfix* padrão do Debian 4.0 (mesmo compilado) roda em *chroot*, enquanto que o *courier-authdaemon* tem proteção de leitura em seu arquivo de *socket* de autenticação, e somente o proprietário e o grupo podem ler o *socket*. Este problema foi detectado analisando os arquivos de *log* do sistema de e-mail (*mail.log e mail.err*). Como solução ao problema do *chroot* foi necessário editar o arquivo *master.cf* do *postfix* para retirar do *chroot* o *daemon* que trabalha com a autenticação SASL. Por fim, foi dada permissão de leitura para o *postfix* no diretório */var/run/courier*, o qual contém o *socket* do

courier-authdaemon. Diversas outras configurações foram feitas no *postfix*, como antivírus, *greylist*, e outros, porém, as únicas problemáticas foram as citadas acima. Com isso, o MTA *postfix* passou a funcionar adequadamente, restando apenas a configuração dos agentes *imap* e *pop.*

Para configurar os agentes imap e pop foi utilizado programa servidor *courier*, que não necessita de *schema* próprio na base LDAP, fazendo uso dos *schemas* padrões e do *schema qmail* utilizado para o *postfix*, tendo sua configuração e *daemons* toda modularizada. O único problema em potencial encontrado com o *courier-authdaemon* versão 0.58-4+etch3 disponível nos repositórios oficiais do Debian 4.0 foi sua limitação quanto ao tipo de *hash* de senhas utilizadas, já que o *courier* consegue trabalhar somente com dois tipos de *hash* de senhas: *clear text* e *crypt.*

O problema potencial é que todos os serviços até então estavam trabalhando com SSHA, padrão este que estava configurado na base LDAP. Assim, o *courier-authdaemon*, módulo responsável pela autenticação, simplesmente não autenticava o usuário, refletindo este problema no *postfix* que utiliza o *courier-authdaemon* para autenticação de SASL. Este problema só pode ser identificado após habilitado o modo *debug* nos *logs* do *courier*, modo este que insere até mesmo as senhas dos usuários nos arquivos de *logs* do sistema, de forma que foi possível observar o erro de autenticação gerado pela falta de suporte ao método criptográfico utilizado (SSHA).

Após a identificação do problema nos arquivos de *logs*, buscou-se a confirmação da limitação nas documentações oficiais do *courier*. Diante da limitação confirmada em documentação oficial surgiram duas escolhas: utilizar *crypt* ou texto plano nas senhas unix da base LDAP. Optou-se pelo uso do *crypt* por questões éticas e de segurança, mas isso demandou alterações no arquivo de configuração do OpenLDAP de forma a ”instruir” o mesmo a forçar senhas *crypt*. Foram necessárias também alterações nos arquivos do *smbldap-tools* de forma que o mesmo trabalhasse com o padrão *crypt* ao invés do SSHA. Adicionalmente, foi ainda necessário que todos os usuários da rede administrativa alterassem suas senhas (no momento da alteração a nova senha é gravada no novo padrão *crypt*) antes de tentar acessar o *e-mail*, do contrário, simplesmente haveria erro na autenticação. É importante frisar que essa alteração de senha foi forçada pelo servidor *samba* administrativo no momento da autenticação do usuário, ou seja, a senha não foi trocada sem o seu conhecimento. A partir daí, a instalação e configuração do *courier* continuou sem problemas.

Nesta fase o ambiente de *e-mail* estava funcional, mas ainda não em produção, pois dois aspectos ainda precisavam ser satisfeitos: a implementação de um *webmail* e o segundo e mais complicado, a migração das caixas de correio atuais no formato *mailbox* para o novo formato *maildir* em outro servidor. No caso do *webmail*, até o momento se tinha como *webmail* oficial o *squirrelmail*. Entretanto, optou-se pelo *roundcube* que também é um *software* livre licenciado sob os termos da GPL e que se apresenta mais interessante visualmente e intuitivo aos usuários. O *roundcube* é desenvolvido em *ajax*e tem a vantagem de armazenar os contatos e preferências dos usuários em base de dados SQL (*MySQL*, *PostgreSQL*ou *SQLITE*), ao contrário do *squirrelmail*, que armazena essas informações em arquivos. Com o conjunto *roundcube*/*OpenLDAP* foi possível interligar as contas de *e-mails* de professores e funcionários diretamente no *webmail*, criando desta forma uma lista centralizada e dinâmica. Para os contatos particulares de cada usuário, o armazenamento é realizado na base SQL (*MySQL*no caso da instituição explorada) juntamente com suas configurações e preferências.

Finalizado a implementação do *webmail*, o ambiente de correio eletrônico estava funcional, restando apenas a questão das caixas de *e-mail* dos usuários que deveriam ser

migradas para o novo ambiente. Certamente nenhum usuário poderia, nem aceitaria, perder seus *e-mails*. O problema a ser resolvido nesta tarefa é migrar caixas de *e-mails* brutas de *mailbox* para *maildir*. Novamente após alguns contatos com colegas da comunidade de *software* livre, chegamos a uma pequena ferramenta chamada *mb2md*e desenvolvida em *perl* e que tem o objetivo de realizar esta migração de formato de caixas, bastando informar a ela os diretórios de origem e destino e alguns poucos parâmetros, conforme a necessidade. Desta forma foi colocado em produção o novo ambiente de correio eletrônico, com o ambiente em produção, iniciou-se a migração das caixas de *e-mail*. Este processo foi realizado com sucesso, migrando todos os e-mails adequadamente.

Todo o ambiente de correio eletrônico estava migrado com uma solução mais flexível, segura e robusta. A ativação de *e-mail* para os usuários passou a ser facultativa, podendo ser habilitada ou desabilitada de acordo com a necessidade da instituição, bem como alteração de cotas, *alias* e-*mails forwarding*. Porém, essas alterações ainda são feitas de forma manual através de arquivos LDIF. Dentro desse novo contexto essas alterações são perfeitamente automatizáveis, não tendo sido ainda prioridade fazê-las.

5.5 Intranet e Sistemas Autenticados

Uma vez que já se tinha uma base de usuários acessados via LDAP, autenticações internas se tornaram bem mais fáceis, como já havia mencionado a linguagem utilizada no desenvolvimento da intranet é PHP, e felizmente o PHP possui uma biblioteca para conexão com LDAP, possibilitando leituras e alterações na base, de forma muito similar a uma base SQL. Desta forma as autenticações de sistemas e intranet passaram a acontecer de forma direta na base LDAP, eliminando a forma pouco comum utilizada no modelo anterior, onde era feita uma autenticação no servidor de *e-mail*.

5.6 *Instant Messenger – Jabber*

O serviço mensageiro também teve um reflexo imediato para os usuários finais, já que foi dada continuidade na utilização da mesma ferramenta para a implementação do serviço *jabber*, o *OpenFire*. Porém, com uma alteração significativa, a integração da autenticação com o restante das aplicações. Esse processo de migração ocorreu de modo facilitado, pois o *OpenFire* apresenta esta possibilidade no momento de sua instalação, e com algumas poucas perguntas a integração do mesmo está completa, bastando para isso que se tenha a base LDAP em funcionamento e com seus usuários e grupos criados.

É importante ressaltar que o *OpenFire* tratará a base LDAP como somente leitura, não realizando nem mesmo alterações de senhas (estas deverão ser alteradas através das estações ou de aplicações específicas, tais como *smbldap-passwd*), sendo que todas as operações de gravação do *OpenFire*, tais como preferências do sistema e do usuários, serão armazenadas na base de armazenamento definida durante a instalação, que no caso da instituição explorada é o MySQL.

5.7 *Samba* Servidor Acadêmico

Vários serviços estavam relacionados ao servidor *samba* acadêmico, tais como *squid*e, *e-mail*, autenticações de estações via *Samba* ou NIS, juntamente com serviço de FTP Acadêmico. O mais importante a ser frisado neste momento é que não há nenhum

serviço administrativo relacionado a este servidor, estando este respondendo apenas por serviços acadêmicos, que se resume basicamente nos citados acima.

Procedeu-se então a configuração do servidor *Samba* em substituição do servidor *Samba* atual. Este novo servidor ganhou um novo hardware, facilitando a sua inserção em produção em paralelo com o antigo. O processo adotado para a configuração/instalação foi exatamente igual ao do servidor *Samba* Administrativo, inclusive com a mesma ferramenta de administração de usuários (o smbldap-tools), o que garante passar pelas mesmas problemáticas e soluções de senhas. Já se conhecia quais os dados que o smbldap-tools migraria automaticamente e quais deveriam ser migrados via *script* manual, assim como o modelo de *hash* de senhas adequado.

Ao término deste processo o ambiente acadêmico estava pronto para realizar autenticações de estações via *Samba* com LDAP. Mesmo assim, o servidor *samba* antigo ainda estava em uso, provendo autenticações a dois serviços, Squid via *Samba* e FTP via NIS. Ambos foram migrados na sequência.

5.8 *Squid* Servidor Acadêmico

Da mesma forma como o servidor *gateway* administrativo, o servidor *gateway* acadêmico passou a ser executado a partir de uma máquina exclusiva. Diferente do cenário original onde se tinha apenas um servidor de *gateway* com *proxy* cache para atender a sub-rede acadêmica e a sub-rede administrativa. Toda autenticação era feita no servidor samba acadêmico.

No novo cenário estes serviços são completamente independentes. Assim, o servidor *gateway squid* acadêmico atende única e exclusivamente a rede acadêmica e se autentica diretamente na base LDAP, eliminando a dependência que existia do samba. Uma vez populadas com usuários, a base LDAP possui integração facilitada com o *squid*.

5.9 FTP

A implementação do serviço de FTP acadêmico também transcorreu de forma tranquila. O software utilizado para prover serviço FTP (vsFTPd) foi substituído pelo ProTFPd. Essa nova instalação ocorreu porque o ProFTPd possui integração facilitada com a base LDAP, alem de ser uma solução estável e amplamente difundida. Mesmo o vsFTPd sendo considerado uma solução segurança, as extensões de segurança não eram utilizadas no cenário original. Ou seja, do ponto de vista de segurança nada mudou, até porque o serviço de FTP não é considerado crítico no âmbito em que é atualmente utilizado na instituição.

O fato é que nesta fase o serviço de FTP está integrado a base LDAP. Sendo este o ultimo serviço que dependia do antigo servidor *samba* acadêmico, com a migração do vsFTPd pode ser desativado. Tínhamos neste momento todo o ambiente corporativo integrado fortemente através de uma solução padronizada, implementada por uma ferramenta livre o OpenLDAP.

5.10 Integrações Faltantes e Previstas

Nesta fase o ambiente está altamente integrado e sem nenhuma dependência do ambiente anterior. Do ponto de vista da equipe de TI, muito havia sido feito, com a

abertura de inúmeras possibilidades de otimização para serem exploradas desse momento em diante. Do ponto de vista dos usuários, pouco ou nada havia sido percebido nas mudanças, com algumas exceções é claro, por exemplo, com o sistema de e-mail. Essa pouca percepção dos usuários reforça a transparência como os processos ocorreram, tanto para os usuários da rede acadêmica como para os usuários da rede administrativa.

Com o ambiente estável e integrado a LDAP, duas novas alterações estão previstas: autenticação de estações linux via *samba* com winbind, eliminando o modo de autenticação paralela que é feito através no NIS. O serviço NIS é extremamente útil e confiável, porém, em nossa realidade atual não é mais interessante ter suporte a dois modos de autenticação de estações (NIS e *Samba*). O fato de escolher apenas um reduz a carga de trabalho e de suporte, juntamente com os problemas potenciais.

Desta forma, num futuro próximo as estações linux estarão realizando suas autenticações via *samba* com winbind e a montagem de diretórios de usuários será realizada via pam mount, eliminando também a necessidade de utilizar exportações via NFS para estações. O NFS, entretanto, continua a ser utilizado somente entre servidores de arquivos, não mais com estações de trabalho.

Autenticações das estações linux no samba acadêmico estão em processo de implantação. A estratégica de migração também deve considerar que é as tarefas sejam realizadas durante o período de recesso acadêmico, diminuindo o risco de afetar usuários.
No samba administrativo nenhuma estação linux realiza autenticação, todas operam com bases locais. Isso não é um reflexo das migrações, pois estas estações nunca realizavam suas autenticações integradas a rede, somente compartilhamento de arquivos. A idéia de alterar este cenário também esta sendo considerada, portanto, será implementada em um futuro próximo.

O potencial de uso do LDAP, o método de acesso a Internet Wireless *WI-FI* também está em fase de implantação. Ao invés do cenário original que é extremamente trabalhoso por exigir configurações manuais em todas as partes do processo, serão feitas pesquisas sobre soluções de *captive portal*[8] para que se realize autenticação dos usuários na base LDAP existente. O único pré-requisito é que a solução adotada seja de software livre, mantendo o histórico de utilização de soluções livres na instituição. A solução de *captive portal* deve apenas solicitar usuário e senha do utilizador, e a partir destes permitir ou não o acesso, sem a necessidade de qualquer configuração manual do administrador da rede ou do usuário, até mesmo o IP deverá ser fornecido via DHCP. O desejável é que este processo seja totalmente automatizado.

Na tabela a seguir pode ser observada a relação de dependência entre serviços no ambiente migrado, já integrado com LDAP, e o *status* da implantação dos serviços.

Serviço	**Serviços Dependentes**	**Status**
LDAP (OpenLDAP)	Nenhum	Implantado
Samba Acadêmico	Servidor LDAP/OpenLDAP	Implantado
Samba Administrativo	Servidor LDAP/OpenLDAP	Implantado

[8] *Captive Portal* – Trata-se de um software responsável por controlar e gerenciar o acesso a internet, de forma automatizada, é muito comum sua utilização em redes publicas a exemplo de aeroportos. Ao digitar o endereço de qualquer site no navegador, o usuário é interceptado pelo sistema *captive portal* e redirecionado para uma interface que solicita suas credencias de acesso.

Servidor de *E-Mail*	Servidor LDAP/OpenLDAP	Implantado
FTP	Servidor LDAP/OpenLDAP, Servidor Acadêmico/NFS	Implantado
Jabber	Servidor LDAP/OpenLDAP	Implantado
Squid Acadêmico	Servidor LDAP/OpenLDAP	Implantado
Squid Administrativo	Servidor LDAP/OpenLDAP	Implantado
Intranet e Autenticações	Servidor LDAP/OpenLDAP	Implantado
Estações Windows	Servidor Acadêmicos e Administrativo/*Samba-winbind*	Implantado
Estações Linux	Servidor Acadêmicos e Administrativo/*Samba-winbind*	A Implantar
Wi-fi	Servidor LDAP/OpenLDAP	A Implantar

Tabela 2 - Relação de Dependência entre Serviços no Ambiente Migrado

6. Conclusão

Observamos a necessidade e a importância de uma centralização dos dados de autenticação para a instituição, e para esta tarefa vimos como opção o protocolo LDAP, implementado pelo software livre OpenLDAP, que vinha de acordo com a filosofia de utilização de software livre na instituição. Esta filosofia norteou não somente a utilização do OpenLDAP, mas todo o processo de migração.

Vimos também que migração de uma ambiente em produção não é tarefa trivial. Ao mesmo tempo é necessária. Algumas questões como a utilização de *hardwares* novos para servidores contribuiu para o processo de migração transparente aos usuários, pois permitiu paralelismo entre serviços velhos e novos quando necessário. Já a dependência entre os serviços do ambiente antigo, obrigou um estudo aprofundado sobre a ordem de migração, de forma a não deixar de oferecer os serviços aos usuários em nenhum momento em que eles necessitassem, garantindo assim a transparência no processo, explorando inclusive o período de recesso acadêmicos para realizar alterações sensíveis a sub-rede acadêmica. Com isso ficou claro que no processo de migração se deve considerar o planejamento de riscos, procurando diminuir surpresas. O primeiro passo é identificar os cenários original e desejável e a partir daí captar os riscos.

Com a necessidade da migração observamos que não necessariamente uma tecnologia tenha de se tornar obsoleta para deixar de atender as necessidades de uma organização. A instituição explorada está em constante evolução e necessita utilizar ferramentas mais flexíveis (escaláveis) para atender satisfatoriamente seus usuários.

Alguns outros itens devem ser pontuados no contexto deste artigo, entre eles a flexibilidade e a variedade de escolhas que se tem para trabalhar com software livre, principalmente quando o assunto é integração e padronização de tecnologias, bem como a histórica robustez dos serviços. Esses atributos justificam não somente este caso de sucesso, mais muitos outros em que o software livre tem um destaque satisfatório. Outra ferramenta extremamente útil e flexível, são os shells de comandos, especialmente o *bash*

do linux, este foi extremamente importante no processo de migração. Grande parte dos trabalhos realizados nas migrações como as senhas NT e LM dos servidores samba e as caixas de *e-mail* foram automatizadas com *scripts bash*, o que o destaca como uma das ferramentas mais importantes desse processo de migração.

A utilização do OpenLDAP permitiu as condições necessárias para que futuramente implementemos diversos outros serviços que operem sobre LDAP, como o *captive* portal e a autenticação das estações Linux sobre o *samba* através do *winbind*, bem como qualquer outro serviço do qual tenhamos necessidade.

Por fim, com o término do final do processo de migração, temos todo o ambiente integrado através de uma ferramenta livre o OpenLDAP, com uma base única de usuários, facilitando os processos de autenticação por se tratar de uma solução padronizada, e temos ainda um ambiente de autenticação mais "limpo", onde deixamos de ter vários serviços interdependentes entre si, e passamos todas as dependências para o LDAP, que desta forma é o único elemento da infra-estrutura necessário a todos os outros serviços.

7. Referências

ABADE, Marcia de Almeida; Integração de Bases de Autenticação LDAP com Bases Corporativas (PostgreSQL) disponível em http://bazar.ginux.ufla.br/index.php/MonosARL/article/view/102/23; acesso em janeiro/2009; Universidade Federal de Lavras: Curso de Administração em Redes Linux, 2004.

AUDY, Jorge Luiz Nicolas; BECKER, João Luiz; FREITAS, Henrique; Modelo de Planejamento Estratégico de Sistemas de Informação: A Visão do Processo Decisório e o Papel da Aprendizagem Organizacional; disponível em http://www.vanti.com.br/ensino/livro/07modelo%20de%20planejamento%20estrategico.PDF; acessado em janeiro/2009; 1999.

BRANDÃO, Wladmir C.; SILVA Antonio Braz de O.; PARREIRAS, Fernando S.; Impactos e Desafios na Implementação de uma Infra-Estrutura de Software Livre nas Organizações: estudo de caso de uma empresa do setor siderúrgico; disponível em http://www.ip.pbh.gov.br/ANO7_N2_PDF/IP7N2_brandao.pdf; acessado em janeiro/2009; Informática Pública vol. 7 (2): 53-66, 2005; 2005.

BUTCHER, Matt; Mastering OpenLDAP: Configuring, Securing, and Integrating Directory Services; Birmingham: Packet Publishing; 2007.

CARTER, Gerald; LDAP System Administration; Sebastopol: O'Reilly; 2003.

CHRISPINIANO. José. Governos Testam Possibilidades do Software Livre. Disponível em: http://www.revista.fundap.sp.gov.br/revista4/paginas/4governo-eletronico.htm

GOVERNO FEDERAL, Grupo de Trabalho de Migração para Software Livre do; Referência de Migração Para Software Livre do Governo Federal; Brasília; 2004.

HOWES, Timothy A., SMITH, Mark C., GOOD, Gordon S.; Understanding and Deploying LDAP Directory Services - 2nd Ed.; Boston: Addison Wesley; 2003.

MACHADO, Erich Soares; MORI, Flavio da Silva Junior; Autenticação Integrada Baseada em Serviço de Diretório LDAP; disponível em http://www.linux.ime.usp.br/~erichsm/mac499/monografia.pdfa cessada em janeiro/2009; Universidade de São Paulo; 2006.

SENA, Clóvis; LDAP: Um Guia Pratico; Rio de Janeiro: Editora Ciência Moderna Ltda., 2005.

TRIGO, Clodonil Honórios; OpenLDAP: uma abordagem integrada; São Paulo : Novatec Editora; 2007.

YEONG, W., HOWES, Timothy A., KILLE, S.; X.500 Lightweight Directory Access Protocol (RFC 1487); IETF Network Working Group; 1993.