UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ – UTFPR
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

MAXIMILLER DOS SANTOS

USO DE LDAP IMPLEMENTADO EM SOFTWARE LIVRE PARA INTEGRAR A AUTENTICAÇÃO DOS CONTROLADORES DE DOMÍNIO MS-ACTIVE DIRECTORY E SAMBA/LINUX

TRABALHO DE DIPLOMAÇÃO

MEDIANEIRA
2013

MAXIMILLER DOS SANTOS

USO DE LDAP IMPLEMENTADO EM SOFTWARE LIVRE PARA INTEGRAR A AUTENTICAÇÃO DOS CONTROLADORES DE DOMÍNIO MS-ACTIVE DIRECTORY E SAMBA/LINUX

Trabalho de Diplomação apresentado à disciplina de Trabalho de Diplomação, do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas – COADS – da Universidade Tecnológica Federal do Paraná – UTFPR, como requisito parcial para obtenção do título de Tecnólogo.

Orientador: Prof. Msc. Paulo Lopes de Menezes

MEDIANEIRA

2013

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Diretoria de Graduação e Educação Profissional
Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

TERMO DE APROVAÇÃO

USO DE LDAP IMPLEMENTADO EM SOFTWARE LIVRE PARA INTEGRAR A AUTENTICAÇÃO DOS CONTROLADORES DE DOMÍNIO MS-ACTIVE DIRECTORY E SAMBA/LINUX

Por

MAXIMILLER DOS SANTOS

Este Trabalho de Diplomação (TD) foi apresentado às 10:20 h do dia 23 de Agosto de 2013 como requisito parcial para a obtenção do título de Tecnólogo no Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, da Universidade Tecnológica Federal do Paraná, *Campus* Medianeira. O acadêmico foi argüido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado com louvor e mérito.

Prof. Msc. Paulo Lopes de Menezes
UTFPR – *Campus* Medianeira
(Orientador)

Prof. Dr. Neylor Michel
UTFPR – *Campus* Medianeira
(Convidado)

Prof. Msc. Ricardo Sobjak
UTFPR – *Campus* Medianeira
(Convidado)

Prof. Msc. Juliano Rodrigo Lamb
UTFPR – *Campus* Medianeira
(Responsável pelas atividades de TCC)

A folha de aprovação assinada encontra-se na Coordenação do Curso.

RESUMO

Com o aumento na quantidade de serviços disponibilizados na Internet, as empresas cada vez mais sentem a necessidade de estar conectadas a rede. Além de um meio para comunicação externa, a grande maioria destas empresas também podem precisar de uma rede que compartilhe os seus recursos locais de software e hardware entre os seus usuários internos de modo que ela tenha controle, garantia e algum nível de segurança dos dados que ali trafegam. Dentre os recursos de infraestrutura pode-se fazer uso do controlador de domínio, através do qual é possível compartilhar e gerenciar os recursos de rede através da criação de um domínio comum que geralmente faz uso de um serviço de diretórios para armazenar os dados deste domínio. O Active Directory e o Samba, escolhidos para este trabalho, são geralmente utilizados para realizar este controle. A nova versão do software livre Samba oferece suporte a esta estrutura de controle de domínio, desta maneira o objetivo deste trabalho é realizar um estudo de integração do Active Directory e Samba dentro de um ambiente simulado e apresentar os procedimentos adotados.

Palavras chave: Tecnologia, Domínio de rede, Protocolo.

ABSTRACT

With the increase in the amount of services available on the Internet, companies increasingly feel the need to be connected to the net. In addition to a means for external communication, the vast majority of these companies might also need a network to share their local software and hardware resources with their internal users so that it has control, and guarantee some level of data security that travels there. Among the infrastructure resources it can make use of the domain controller, through which it is possible to share and manage network resources through the creation of a mutual domain which usually makes use of a directory service to store the data in this field. Active Directory and Samba, chosen for this work, are generally used to perform this control. The new version of Samba free software supports to this control structure domain, this way, the goal of this work is to perform a study of integration of Active Directory and Samba in a simulated environment and show the procedures adopted.

Keywords: Technology, Domain Network, Protocol.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE ABREVIATURAS

AD	*Active Directory*
ARP	*Address Resolution Protocol*
BDC	*Backup Domain Controller*
CIFS	*Common Internet File System*
DHCP	*Dynamic Host Configuration Protocol*
DNS	*Domain Name System*
FTP	*File Transfer Protocol*
LDAP	*Lightweight Directory Access Protocol*
MIT	*Massachusetts Institute Of Technology*
NetBEUI	*NetBIOS Extended User Interface*
NetBIOS	*Network Basic Input/Output System*
NFS	*Network File System*
NTP	*Network Time Protocol*
PDC	*Primary Domain Controller*
RPC	*Remote Procedure Call*
SMB	*Server Message Block*
SNTP	*Simple Network Time Protocol*
TCP	*Transmission Control Protocol*
TI	*Tecnologia da Informação*
WINS	*Windows Internet Name Service*
XDR	*External Data Representation*

SUMÁRIO

1 INTRODUÇÃO

Os profissionais de TI (Tecnologia da Informação) por tempos almejam por colocar o Active Directory para trabalhar em conjunto com o Samba, interagindo de forma transparente para realizar atividades centralizadas como incluir, editar e excluir usuários e grupos, adicionar um usuário a grupos, retirar usuários de um grupo, gerar políticas de grupos. Com isto espera-se um ganho considerável em qualidade e desempenho.

Além de ser uma dentre as poucas ferramentas disponíveis no mercado que desempenha estas funções – até o presente momento – a Microsoft não se sente ameaçada pelo concorrente, agora direto, Samba 4. A própria empresa contribuiu com algumas linhas de seu código e desenvolvedores para que o projeto realizasse grandes avanços. Parece muito promissor, já que se baseia na mesma estrutura e trabalha com os mesmos protocolos do Active Directory a nova versão do Samba pode replicar a base de dados e aplicar também configurações.

Desta forma este trabalho tem como intuito implantar o Active Directory integrando-o com o Samba em sua nova versão e mostrar como este será um grande salto dentro da tecnologia de infra-estrutura e de controladores de domínios de redes.

1.1 OBJETIVO GERAL

Implementar um método de integração entre o Active Directory e o Samba.

1.2 OBJETIVOS ESPECÍFICOS

- Implantar os serviços Active Directory e Samba, em um ambiente virtualizado;
- Configurar os servidores Active Directory e Samba para trabalhar integrados em um domínio;
- Ingressar e gerenciar máquinas clientes e usuários no domínio;

- Constatar a replicação automática dos dados entre o Active Directory e o Samba.

1.3 JUSTIFICATIVA

Versões anteriores do Samba não atendem as expectativas em termos de integração e autenticação com o Active Directory. Esperada há tempos, a integração destes controladores pode se tornar realidade com a versão do samba, que ainda encontra-se em fase de desenvolvimento, a qual teve início em 2005 e já possui uma versão estável. O que se busca é ter todos os serviços centralizados em um único programa, desta forma ganhando desempenho e qualidade no serviço em caso de replicação. Uma parte que chama atenção deste produto para o mercado corporativo e que pode alavancá-lo é o fato de ser uma ferramenta de código aberto, ou seja, não agregando custo de licenciamento de *software*.

1.4 ESTRUTURA DO TRABALHO

Capítulo 1: Apresenta a introdução, objetivos gerais e específicos, bem como, a justificativa.

Capítulo 2: Apresenta a fundamentação teórica e conceitos dos protocolos e serviços envolvidos nos controladores de domínio.

Capítulo 3: Apresenta as ferramentas, programas, métodos utilizados para a implementação dos testes, problemas conhecidos e soluções.

Capítulo 4: Apresenta a conclusão do trabalho, bem como as considerações finais e trabalhos que podem ser desenvolvidos futuramente com base neste trabalho.

2 REVISÃO BIBLIOGRÁFICA

2.1 SERVIÇO DE DIRETÓRIOS

Os diretórios de uma maneira geral, servem para agilizar o processo de pesquisa de informações. Estas informações podem ser dados de usuários ou informações sobre recursos de rede. Diretório é um serviço de armazenamento de informações, que atua de forma hierárquica para aperfeiçoar a leitura, pesquisa e busca de dados, tornando assim mais ágil e fácil a inserção de objetos no diretório. Para facilitar, dois exemplos práticos de diretórios usados diariamente sem percepção comum são agenda e lista telefônica. Ambos guardam informações para busca, em ordem alfabética deixando assim mais rápida a consulta (TRIGO, 2007, p.239).

De uma forma simplificada pode-se dizer que um diretório é um banco de dados simples, onde são armazenadas informações de usuários e devido a sua forma hierárquica são rapidamente encontradas. Entretanto as operações realizadas em um banco de dados relacional são mais complexas do que as realizadas em uma base de dados para diretório. Nos bancos de dados tradicionais os dados são armazenados em tabelas e em um diretório as informações ou atributos de um usuário são organizados em uma estrutura de árvore (TRIGO, 2007, p.239). Por exemplo, uma busca no diretório da cidade *A* homens entre 35 e 40 anos de idade e que sejam divorciados.

Um serviço de diretório precisa ser extensível, podendo servir a diferentes propósitos, por exemplo, armazenar os telefones de lojas da cidade A, mas aceitar também endereços como atributos, o propósito pode ser estendido acoplando-o a um guia de ruas.

Um diretório além de conter registros pode também conter outros diretórios e assim sucessivamente. Esse diretório dentro de outro diretório é chamado de subdiretório. Essa organização cria uma hierarquia de diretórios ou árvore de diretórios no sistema operacional.

A Figura 1 ilustra a estrutura de um diretório.

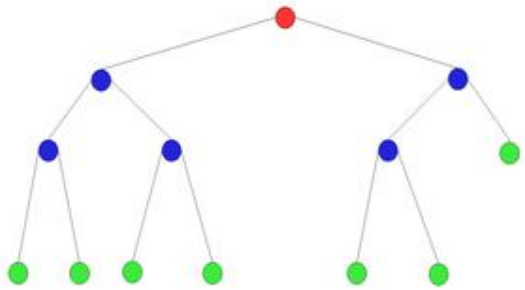

Figura 1 - Representação de uma hierarquia de diretórios.
Fonte: (SCRIMGER et al, 2002, p.642).

Os diretórios possuem nós e são organizados em níveis, como mostra a Figura 1. O nó mais alto – na cor vermelha – é chamado de nó raiz, e seus subdiretórios chamados de nós - filhos e possuírem diretórios abaixo dele – em cor azul – ou folhas caso sejam o último diretório da hierarquia – em cor verde. Dessa maneira se obtêm um diretório centralizado e organizado, composto por uma maneira eficiente de consultar as informações.

2.2 CONTROLADOR DE DOMÍNIO PRIMÁRIO

Um controlador de domínio primário (PDC) é uma máquina servidora dentro de um domínio. Um domínio por sua vez é composto por computadores agrupados abaixo dele, com o propósito de controlar os usuários, permitir ou negar acesso a recursos da rede, por exemplo, desativar o painel de controle de uma estação Microsoft Windows. Cada domínio pode ser chamado de árvore, um ou mais domínios são chamados de florestas. Estas florestas são controladas pelo PDC (MINASI et al, 2000).

Uma rede pode ter somente um controlador de domínio primário, porém outros servidores podem ser configurados na mesma rede, para dividir a tarefa de controlar o domínio e serão chamados de controladores de domínio *backup* (BDC). Estes podem também autenticar usuários, contudo as atualizações só podem ser

feitas no PDC que posteriormente irá replicá-las por toda a rede. Em ambientes de médio e grande porte, ambos podem trabalhar de forma conjunta para que o controlador de domínio primário não seja sobrecarregado. Outra vantagem desta estrutura é a redundância da informação entre os servidores, de forma que um BDC se torna um servidor de *backup*, que pode vir a assumir o papel de PDC, caso o servidor principal venha se tornar indisponível por alguma razão (MINASI et al, 2000).

2.3 MICROSOFT ACTIVE DIRECTORY

O Microsoft Active Directory (AD) é um banco de dados para armazenamento de informações sobre usuários, máquinas e informações administrativas. A base de dados Active Directory recebe o nome de NTDS.DIT e tem grande capacidade de armazenamento de recursos. Essa base de dados é organizada de forma hierárquica aplicada através do protocolo Ldap (*Lightweight Directory Access Protocol*) e está disponível em servidores da família Microsoft a partir da versão Windows Server 2000 (MINASI et al, 2000).

Todos esses objetos como são chamados de recursos, como contas de usuários e computadores, grupos de usuários e grupos de computadores ficam armazenados na base de dados NTDS.DIT que é responsável pela organização da base Ldap do servidor. Outros serviços de rede como DHCP (*Dynamic Host Configuration Protocol*) e DNS (*Domain Name System*) são implementados junto com o Active Directory para gerenciamento de endereço IP e resolução de nomes de máquinas clientes e servidores.

Além do DHCP e do DNS, o protocolo Kerberos também está presente e tem como funcionalidade a autenticação criptografada dos dados, elevando a confiabilidade do serviço (MINASI et al, 2000).

2.4 SAMBA

O Samba é um programa desenvolvido para servidor GNU/Linux – e sistemas baseados em Unix – que implementa o protocolo SMB (*Server Message*

Block) e da suporte ao gerenciamento e compartilhamento de recursos de redes compostas por computadores Windows e obviamente GNU/Linux. Assim, é possível usar o Linux como servidor de arquivos, servidor de impressão e também como PDC, da mesma forma como se estivesse sendo utilizados servidores Windows (COSTA, 2010).

2.4.1 HISTÓRIA DO SAMBA

Inicialmente Andrew Tridgell, australiano desenvolvedor do Samba, precisava montar em uma máquina com sistema operacional DOS, com sistema de arquivos NFS (Network File System), um espaço no disco rígido para seu servidor Unix, porém carecia de suporte ao protocolo NetBEUI, não suportado pelo NFS (COSTA, 2010).

Tridgell desenvolveu um *sniffer*[1] capturou e analisou os pacotes que foram transmitidos de forma aleatória em sua rede, aplicou engenharia reversa em cima do protocolo SMB e implementou-o no Unix. Assim sua máquina com sistema operacional DOS passou a responder as requisições realizadas pelo servidor Unix como se os sistemas operacionais fossem iguais em ambos os lados. Em 1992 o código foi disponibilizado publicamente por Andrew, mas o projeto não foi levado adiante.

Em 1994 a Microsoft publicou as especificações dos protocolos SMB e NetBEUI, o que fez com que Andrew vendo a documentação atualizada e liberada de forma completa voltasse a desenvolver e aprimorar o que havia começado anos atrás. O desempenho do Samba, tanto como servidor de arquivos ou PDC, é considerado uma boa alternativa *open source* para o Active Directory.

2.5 ACTIVE DIRECTORY E SAMBA

A integração de sistemas operacionais distintos está na criação de condições favoráveis para que os sistemas possam se comunicar. A necessidade de

[1] Sniffer: Programa utilizado para capturar os dados que trafegam pela rede.

integrar sistemas originou-se diretamente da utilização generalizada das redes de computadores. Independente de qual seja a forma de integração é indispensável que haja algo em comum entre eles, algo que faça parte de ambos os sistemas, por exemplo, um protocolo de rede.

Até o início do século XXI a grande diferença entre os sistemas e as aplicações que eram executadas em rede era sobre qual protocolo seria usado, mas isto mudou, utiliza-se o protocolo de rede TCP (*Transmission Control Protocol*) e os problemas são específicos, relacionados com estrutura ou arquitetura.

Se os desenvolvedores do Samba desejavam uma maior integração entre servidores GNU/Linux e Microsoft Windows um grande passo foi remanejar a estrutura e integrar ao próprio Samba protocolos que também eram usados pelo Active Directory.

No Quadro 1 pode ser visto os protocolos que complementam os serviços.

PROTOCOLOS	DESCRIÇÃO
SMB/CIFS	Protocolo relacionado ao compartilhamento de arquivos
LDAP	Protocolo utilizado para implementar o Active Directory
DNS	Protocolo responsável pela resolução de IP em nome de máquinas e vice-versa.
KERBEROS	Protocolo para autenticação criptografada
NTP	Protocolo para sincronização entre as máquinas
DHCP	Protocolo responsável pela distribuição de endereço IP

Quadro 1 - Protocolos comuns entre os controladores de domínio.
Fonte: (MICROSOFT, 2013b).

O uso do protocolo Ldap favoreceu para que isto fosse possível, pois devido ao modo como este trabalha, é possível ter os utilizadores em registros únicos, contendo os campos necessários tanto para o Samba como para o Unix.

2.5.1 *SERVER MESSAGE BLOCK* – SMB

Quando se fala em compartilhamento não se refere somente a pastas e arquivos, com o protocolo podem ser compartilhados também dispositivos, por exemplo, portas seriais e impressoras. É importante saber que SMB é um protocolo e não um programa.

Desenvolvido no início dos anos 90 por Dr. Barry Feigenbaum, na época um funcionário da IBM, este protocolo foi criado inicialmente para rodar em cima da API NetBIOS/NetBEUI, mas desde o Windows 2000 é executado em cima do protocolo TCP (SHARPE, 2002).

O uso mais comum deste protocolo é o compartilhamento de arquivo dentro de uma rede ou domínio, onde um cliente acessa um servidor e utiliza-se de seus objetos – cliente e servidor podem ser quaisquer computadores que tenham o protocolo ativo (SHARPE, 2002).

O protocolo SMB trabalha com o envio de pacotes entre os *hosts* envolvidos, onde cada pacote transmitido inclui algum tipo de requisição, por exemplo, ler, salvar, abrir ou fechar um arquivo. Por sua vez, o computador que recebeu a requisição verifica se esta é válida – se o arquivo solicitado existe ou se o usuário tem permissão para acessá-lo – e se for verdadeira a executa e envia uma resposta para o *host* solicitante (SHARPE, 2002).

A razão principal que fez o protocolo SMB receber contínuas modificações foi a falta de suporte a autenticação. Qualquer usuário poderia escrever e ler tudo que estava a seu alcance, o que não era muito adequado para grandes empresas. A necessidade de separação e estabelecer privilégio aumentaram então mais tarde a Microsoft estendeu o protocolo SMB para incluir suporte a gerenciamento de rede e outros serviços. O protocolo que originalmente era apenas para compartilhamento incorporou outras funcionalidades e foi renomeado para CIFS.

2.5.2 COMMON INTERNET FILE SYSTEM – CIFS

O protocolo CIFS é uma extensão melhorada do protocolo SMB e disponibiliza acesso transparente de arquivos compartilhados em rede, foi criado pela Microsoft para trabalhar em qualquer tipo de plataforma, ou seja, desenvolvido

para trabalhar independentemente da arquitetura de rede, protocolo de transporte e sistema operacional. Esta independência é alcançada pelo uso do RPC (*Remote Procedure Call*) onde seus protocolos são descritos utilizando a XDR (*External Data Representation*) (UFRJ, 2009).

A RPC é uma biblioteca de procedimentos com o objetivo de permitir que um usuário – cliente – possa utilizar um aplicativo executando uma chamada a outro usuário – servidor – de forma que pareça que o mesmo esteja em seu próprio espaço (UFRJ, 2009).

A XDR é um protocolo para apresentar dados. Realiza transferência de informações entre máquinas de arquiteturas e sistemas operacionais diferentes. Desta forma é utilizado pelo RPC para resolver problemas de compatibilidade entre máquinas e sua forma de interação (UFRJ, 2009).

O protocolo CIFS também trabalha com o envio de pacotes entre os *hosts* envolvidos, onde cada pacote transmitido inclui algum tipo de requisição, como por exemplo, utilizar uma impressora. O computador que recebeu a requisição verifica se esta é válida – se a impressora solicitada existe e está disponível – se for verdadeira a executa e envia uma resposta para o *host* solicitante. Além da idéia principal do protocolo CIFS ser o compartilhamento de arquivos existem mais funções associadas a ele, podendo definir níveis de segurança e autenticação para acesso de dados e outras funções que podem ser realizadas através do PDC. Os compartilhamentos existentes no servidor sofrem restrições de acesso e são liberadas depois que o usuário se autentique com *login* e senha. Para a parte de autenticação a senha é enviada de forma criptografada.

Diferente do protocolo SMB, o protocolo CIFS suporta múltiplas requisições, isso é feito através do uso de um id multiplexada (MID). Cada uma das requisições enviadas ao servidor possui um único MID e essas requisições são respondidas com o mesmo MID. Assim, múltiplas requisições podem ser enviadas ao servidor e quando o cliente receber a resposta precisará comparar os MID para saber qual requisição foi respondida.

2.5.3 PACOTES SMB/CIFS

Os pacotes enviados e recebidos através das requisições e respostas realizadas pelo protocolo SMB/CIFS possuem o cabeçalho conforme a Figura 2 para não se perderem quando forem transmitidos (UFRJ, 2009).

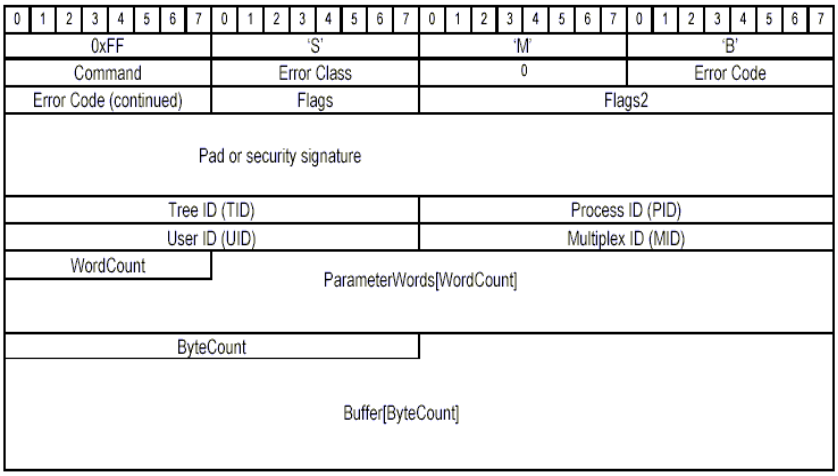

Figura 2 - Cabeçalho de pacotes SMB/CIFS.
Fonte: (UFRJ, 2009).

Cada pacote independente se está sendo enviado ou recebido contém um cabeçalho de 4 octetos. O primeiro octeto é 0xFF, logo em seguida uma representação ASCII das letras 'S', 'M' e 'B'.

Command: este campo contém um código e seu tamanho é de 1 octeto informando o tipo de pacote.

Error class: se a requisição enviada foi recebida pelo servidor o valor é zero, indicando sucesso, mas o campo pode também indicar outros valores, indicando informações diferentes como mostra os valores abaixo:

- ERRDOS (0x01) – Erro do núcleo de instruções do sistema operacional DOS;
- ERRSRV (0x02) – Erro é gerado pelo gerenciador de arquivos de rede do servidor;
- ERRHRD (0x03) – Erro no *hardware*;
- ERRCMD (0xFF) – O comando não estava no formato 'SMB'.

Error code: este campo contém 16 octetos e mostra o tipo de erro ocorrido, recebendo valor zero se não houver erro. Se receber este número juntamente com o *error class* indica que erro aconteceu, os mais comuns são *bad password* ou *file does not exist*. Da mesma forma que o *error class* somente é configurado em uma resposta as requisições pelo servidor.

Flags: os 8 octetos especificam opções particulares do pacote, exceto o bit 3 se for configurado indica que não se deve levar em consideração a diferença entre caracteres maiúsculos e minúsculos.

Flags2: Mais opções como segue abaixo:

- bit 0 - o servidor pode retornar arquivos com nomes longos;
- bit 16 - as strings no pacote estão codificadas com Unicode;
- bit 6 - indica que qualquer caminho na requisição pode ser um arquivo com nome longo.

Tree ID (TID): formado por 2 octetos para mostra qual recurso a solicitação está se referindo, caso não haja a solicitação de nenhum recurso o campo é ignorado.

Process ID (PID): ocupa 2 octetos para identificar qual processo realizou a requisição. Este identificador é usado para resolver problemas de concorrência pelo servidor.

User ID (UID): número de 2 octetos que identifica qual usuário está fazendo as requisições na máquina cliente. O cliente obtém um UID e senha com o qual poderá fazer requisições, após se identificar com os dados e ser verificado se o mesmo tem permissão para acessar arquivos e impressoras senão o servidor envia uma mensagem dizendo que este UID não possui tais privilégios.

Multiplex ID (MID): número formado de 2 octetos gerencia múltiplas requisições. Quando um cliente faz uma requisição ao servidor, ele verifica o MID para ver se tem alguma requisição pendente.

WordCount e ParameterWords: são armazenados os dados específicos do comando. O campo *WordCount* indica quantas palavras de 2 octetos o campo *ParameterWords* contém. Assim cada pacote se encaixa ao tamanho certo para transmitir dados de seu comando.

ByteCount e buffer: parecidos com o *WordCount* e *ParameterWords* armazenam uma quantidade de dados variável específica numa base por pacote.

2.5.4 DOMAIN NAME SYSTEM – DNS

O DNS é uma base de dados distribuída utilizada para resolver nomes de domínios em endereços IP ou vice-versa e realiza um mapeamento dos nomes de domínios e seus respectivos endereços IP. Comumente usado na Internet em que todas as máquinas possuem um endereço IP (SCRIMGER et al, 2002, p.642).

Por exemplo, o endereço *www.meudominio.com.br* está ligado ao endereço IP *200.200.210.150*. A resolução deste endereço se dá da seguinte maneira:

- O nome do *host* é inserido no navegador, *prompt* de comando ou outro serviço;
- O sistema operacional verifica se o nome da máquina de destino é o mesmo configurado localmente;
- Se não for correspondido o resolvedor – o cliente – envia uma requisição ao servidor DNS e se este encontrar uma resposta envia o número de IP ao solicitante. Caso não houver uma resposta á solicitação, uma mensagem de erro será exibida para o usuário.

Um servidor DNS é composto por três elementos:

- Espaço de nome: Ambiente de nome da Internet ou uma área de nome interno definido conforme a necessidade.
- Resolvedores: Clientes ou locais de onde surgem as solicitações para a resolução de nomes. Estes enviam ao servidor DNS as solicitações para conversão e podem ser desde estações de trabalho até mesmo outros servidores.
- Servidor de nomes: Computador que possui uma aplicação do servidor DNS e responde as solicitações dos clientes.

Na Figura 3 está apresentada a estrutura de um servidor DNS que é baseada no conceito de espaços de nomes e árvore de domínios.

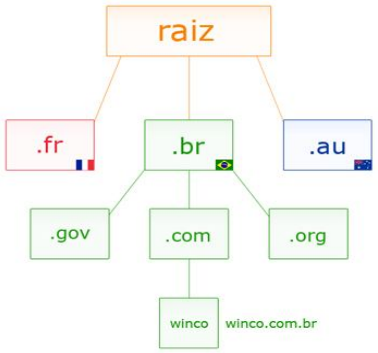

Figura 3 - Estrutura de um servidor DNS.
Fonte: (WINCO, 2013).

A raíz, chamada comumente de *root*, é o domínio de mais alto nível sendo representada por um ponto, seguida pelos nomes de domínios de níveis mais baixos.

Na Internet, o DNS tenta resolver o endereço *www.empresa.com.br* digitado em um navegador, caso não tenho sucesso outro servidor DNS será chamado de tempo em tempo para efetuar a resolução do nome em endereço IP e assim sucessivamente. Entretanto na rede interna o servidor DNS é utilizado para localizar recursos dentro da mesma, por exemplo, pode se utilizar uma máquina que tenta se conectar ao endereço *servidor**pastadousuario*, para isso é realizada uma consulta ao servidor DNS, que poderá retornar o endereço IP de servidor, caso seja encontrado senão uma mensagem de erro é enviada (SCRIMGER et al, 2002, p.642).

2.5.5 LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL – LDAP

Formado por um conjunto de protocolos cliente/servidor, utilizado para acessar um determinado diretório e suas informações, permitindo navegar, ler e consultar atributos. Diferente de um banco de dados relacional qualquer cliente Ldap pode obter dados de um servidor Ldap já que não precisa de nenhuma biblioteca especifica para implementar este protocolo. A forma de armazenamento dos dados e o sistema operacional base não fazem parte do protocolo (TRIGO, 2007, p.239).

O Ldap trabalha de forma transparente, fornecendo também um acesso rápido e fácil, sendo padrão em serviços de diretórios já que pode ser usado em diferentes plataformas (SCRIMGER et al, 2002, p.642).

As informações da fonte de dados são armazenadas em uma base de dados, e organizada de forma similar ao DNS, feita de forma hierárquica partindo da raiz e chegando, por exemplo, até a impressora ou servidor (ERICH, 2005).

Um servidor Ldap é responsável pela autenticação do usuário na rede e as informações deste usuário ficam armazenadas na base de dados do servidor e o mesmo permite ou não que o cliente realize consultas e modificações. Permitindo a consulta de informações pode também ser utilizado como agenda de contatos, no qual foi utilizado inicialmente (ERICH, 2005).

Sendo um sistema centralizado a manutenção de um cadastro torna-se mais difícil para o administrador, devido à rotatividade de clientes. Como vantagens do Ldap citam-se (THE OPENLDAP FOUNDATION, 2003):

- As implementações podem ser feita de diferentes maneiras trazendo novas interfaces e ferramentas de administração e consulta, mas a forma básica de operar é definida no protocolo;
- Por ser um protocolo de código aberto, a utilização entre diferentes fornecedores é facilitada. Um cliente Ldap baseado em *OpenLDAP* pode realizar consultas e atualizações em um servidor de outro fornecedor que segue padrões Ldap.
- API bem definida e com suporte para diversas linguagens de programação;
- Muito mais rápido que sistemas de bancos de dados tradicionais, levando em conta que as atualizações são menos constantes que consultas;
- As regras de armazenamento de dados são padronizadas e existem para diferentes funções;
- Facilmente replicável e distribuído.

2.5.5.1 Serviço de rede do LDAP

Como o Ldap é um serviço de autenticação e armazena as informações em um diretório centralizado, outros serviços de rede como, por exemplo, o Samba até a versão 3.6.9 precisa ser configurado de maneira que não realize a busca em seus próprios arquivos de textos.

O Ldap foi desenhado para ser escalável a milhões de objetos, utilizando mecanismos de replicação e de distribuição das informações. Uma rede pode ter vários servidores Ldap, cada qual responsável por uma parte da árvore do diretório e cada um desses servidores pode ter vários servidores de *backup*, garantindo o desempenho e a tolerância às falhas do sistema como um todo. Um mesmo servidor Ldap pode ser o primário para determinado serviço da árvore e secundário para outro, permitindo a distribuição e o acesso eficiente das informações

Este protocolo possui um grau elevado de segurança. A troca de informações entre um servidor Ldap e um cliente pode ser autenticada por certificados digitais utilizando outro protocolo de rede chamado Kerberos, detalhado mais adiante, como mecanismo de autenticação distribuída.

2.5.6 KERBEROS

Este protocolo foi criado pelo MIT (*Massachusetts Institute Of Technology*) na década de 80 e sua versão atual é a Kerberos 5. Segue Abaixo algumas características da autenticação do Kerberos (CONECTIVA, 2009):

- *Single sing-on:* A senha é solicitada para o usuário somente uma vez. Se algum outro serviço que necessite de autenticação for solicitado, a senha não precisará ser informada novamente;
- Senha criptografada*:* A senha sempre é codificada antes de ser transmitida pela rede;
- Autenticação centralizada*:* Uma mesma senha pode ser utilizada para vários serviços tendo em vista a centralização da base de dados, o que facilita a memorização e definição de políticas de segurança globais;
- Redundância: A autenticação é feita utilizando-se mais de uma fonte;

- Múltiplos domínios: Usuários de domínios diferentes podem se autenticar entre si;
- Fácil configuração do cliente*:* A configuração é inserida no DNS, fornecendo na grande maioria a estrutura necessária para a utilização do Kerberos;
- Padronização: Por este motivo aplicações diferentes podem conversar, como Microsoft Windows 2008 e o GNU/Linux, tendo em vista que ambos usam Kerberos, onde máquinas Microsoft podem obter *tickets* de máquinas GNU/Linux ou vice-versa (CONECTIVA, 2009).

O Kerberos realiza a autenticação da máquina cliente no servidor, trata de forma segura a aplicação que oferece o serviço. Trabalha com *tickets*, que servem para certificar a autenticidade do usuário e garantir o acesso as aplicações e serviços disponibilizados pelo servidor. Quando um usuário entra com as informações de *login*, os dados são enviados para este servidor que os recebe e confere com as informações que estão cadastradas. Essas informações são criptografadas com a própria senha do usuário e enviadas para o cliente. Se as informações do *ticket* forem descriptografadas o usuário é quem diz ser. O certificado é armazenado na máquina cliente e por questões de segurança possui tempo de vida útil, caso tenha sido interceptado durante a troca de mensagens. Quando um cliente faz a solicitação de um serviço, numa segunda vez, utiliza o certificado e somente será preciso se autenticar novamente se este estiver com a vida útil vencida. O servidor Kerberos realiza verificações e responde com um *ticket* chamado de TGS (*Ticket-Grant-Service*), necessário para solicitação dos serviços. É importante saber que para utilizar mais de um serviço, deve ser gerado um novo *ticket* TGS, por exemplo, se um cliente solicitar um serviço *web* e outro de *e-mail*, este cliente deve se autenticar ou enviar um ticket com vida útil válida para cada serviço solicitado (FILHO, 2009).

A Figura 4 ilustra o processo de autenticação pelo Kerberos.

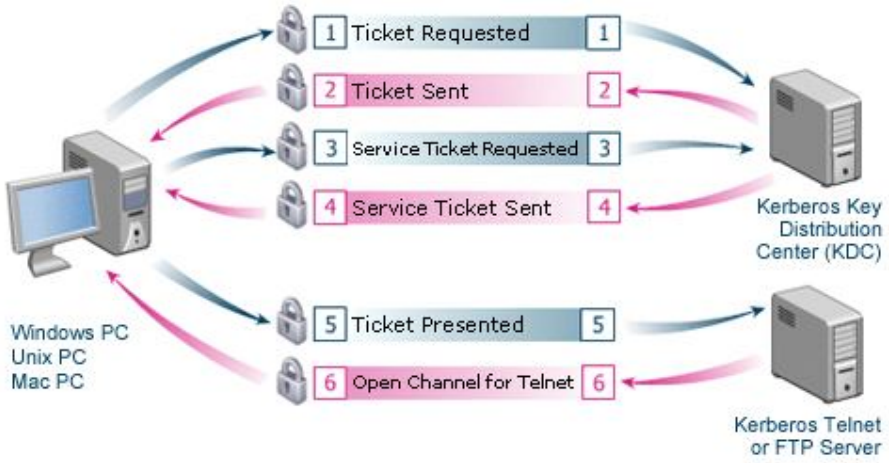

Figura 4 - Autenticação Kerberos.
Fonte: (ERICOM, 2013).

2.5.7 NETWORK TIME PROTOCOL – NTP

Este protocolo é utilizado para a sincronização de relógios entre computadores dentro de uma rede local ou Internet (NTP, 2013).

O não sincronismo dos relógios de uma rede pode causar problemas de administração, por exemplo, em:

- Servidores de controle de versão;
- Sistemas de *backup*;
- Transações de banco de dados;
- Erros no Active Directory;
- Análise de *log* de vários servidores;
- Servidores de *e-mail*.

O SNTP (*Simple Network Time Protocol)* é uma adaptação do NTP feita pela Microsoft. Este protocolo é utilizado para manter o sincronismo entre as estações de trabalho e o servidor, onde a exatidão do NTP não é tão exigida. (MICROSOFT, 2009a).

O protocolo Kerberos utiliza este protocolo para auxiliar na autenticação das máquinas dentro da rede. Possui uma tolerância e aceita como margem de erro uma diferença de no máximo 5 (cinco) minutos entre cliente e servidor.

2.5.8 DYNAMIC HOST CONFIGURATION PROTOCOL – DHCP

Protocolo que possibilita uma máquina adquirir um endereço IP de forma dinâmica e outras configurações, como servidor DNS e gateway dentro de uma rede. Este servidor realiza o gerenciamento de uma faixa de endereços válidos, chamado escopo de rede. Essa distribuição coloca um prazo de validade em cada endereço IP liberando-os quando possível, esse processo recebe o nome de concessão (SCRIMGER et al, 2002, p.642).

Quando uma máquina é iniciada na rede, esta emite um sinal, uma mensagem chamada de *DHCPDISCOVER* dentro da rede por *broadcast*, está mensagem chega ao servidor DHCP o qual recebe essa mensagem e envia uma resposta oferecendo um endereço IP e informações complementares para o cliente, juntamente com as informações de concessão, essa resposta é chamada de *DHCPOFFER*. Após isso o cliente envia uma mensagem solicitando as configurações oferecidas pelo servidor, chamada de *DHCPREQUEST*, com tudo preparado para a solicitação o servidor então manda uma mensagem de reconhecimento e permissão para a utilização das informações cedidas, esta ação recebe o nome de *DHCPPACK*. Quando o cliente recebe a permissão entra no estado limite, onde utiliza temporizadores para controlar o vencimento, a renovação e a revinculação da concessão. Quando o tempo de concessão expira o cliente emite uma mensagem *DHCPREQUEST* ao servidor que concedeu o endereço IP entrando no estado de renovação e aguardando uma resposta do servidor que responde aceitando a solicitação com o envio do pacote *DHCPPACK* ou rejeitando com o pacote *DHCPNACK*. Caso a solicitação seja rejeitada o endereço IP é liberado e volta para o estado de inicialização. Sendo negada a renovação de endereço IP o cliente entra em estado de revinculação e retransmite a mensagem *DHCPREQUEST* para o servidor, se receber a mensagem *DHCPPACK*, volta para o estado limite caso contrário retorna para o estado de inicialização (SCRIMGER et al, 2002, p.642).

Na Figura 5 pode ser vista uma troca de mensagens entre cliente/servidor.

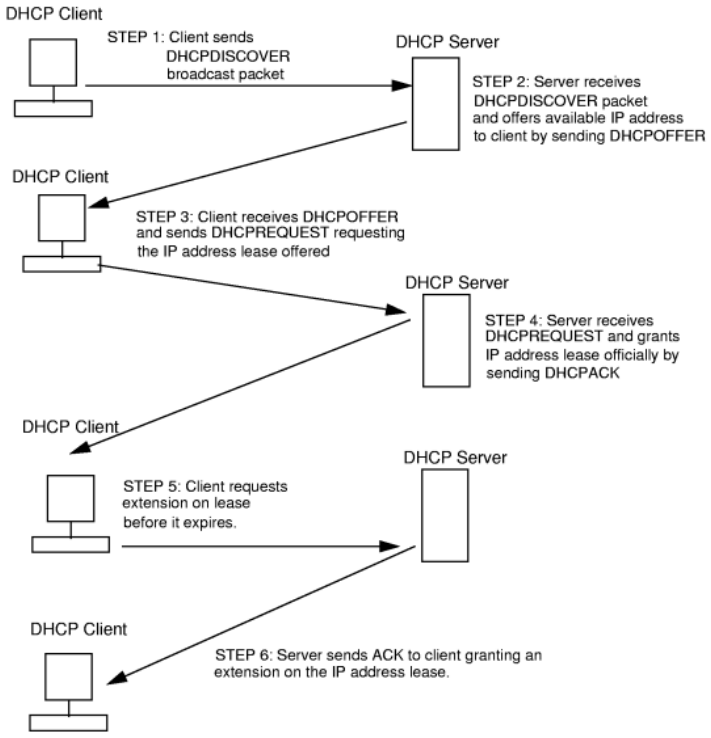

Figura 5 - Troca de mensagem entre cliente e servidor DHCP.
Fonte: (LEVIS, 2013).

2.5.9 NETBEUI E NETBIOS

Protocolos desenvolvidos pela IBM na década de 80 para dar suporte ao gerenciamento da rede.

O NetBEUI pertence à Microsoft e é uma extensão do protocolo original NetBIOS. Este protocolo é divido em I-Frame – Frames de informações numéricas, utilizados para fluxo de dados em sequência – e UI - Frame – Frames de informações não numéricas que fornecem datagramas (UFRGS, 2013).

O NetBIOS é um protocolo de alto nível que inicialmente atuava com o sistema operacional DOS e sofreu alterações com o tempo adquirindo interface para usuário, recebendo o nome de NetBEUI. Porém mesmo após esta mudança é utilizado o termo NetBIOS para referenciar ambos de forma genérica.

O NetBIOS foi desenvolvido com a intenção de permitir a comunicação entre máquinas dentro de uma rede. Nesta estrutura foi implementado o conceito de nome

de serviço, o que tornou possível conectar-se a outro computador com nome reservado. Não é obrigatório um servidor centralizado o NetBIOS possuí três métodos de resolução de nomes:

- *Broadcast* NetBIOS: Funciona basicamente como o protocolo ARP (*Address Resolution Protocol*) onde o cliente NetBIOS envia uma pergunta para todo o domínio de broadcast e aquele que possuir a resposta envia os dados corretos para a origem da pergunta. A Figura 6 ilustra a forma de comunicação.

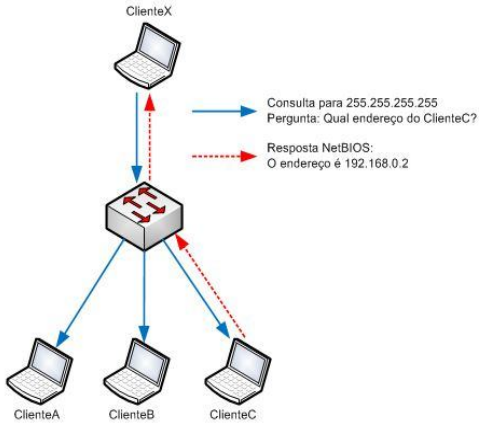

Figura 6 – Funcionamento de broadcast NetBIOS.
Fonte: (Romero, 2011).

- *WINDOWS INTERNET NAME SERVICE* – WINS: O WINS é um serviço que essencialmente consiste numa lista de nomes de computador e seus respectivos endereços IP. A vantagem do WINS sobre o método de *broadcast* é o fato de este permitir que as consultas ultrapassem os limites do domínio de *broadcast* local. Ao habilitar um cliente para fazer consultas em um servidor WINS automaticamente estará o adicionando à lista do serviço WINS. Na Figura 7 é demonstrado o funcionamento de um servidor WINS (MICROSOFT, 2009c).

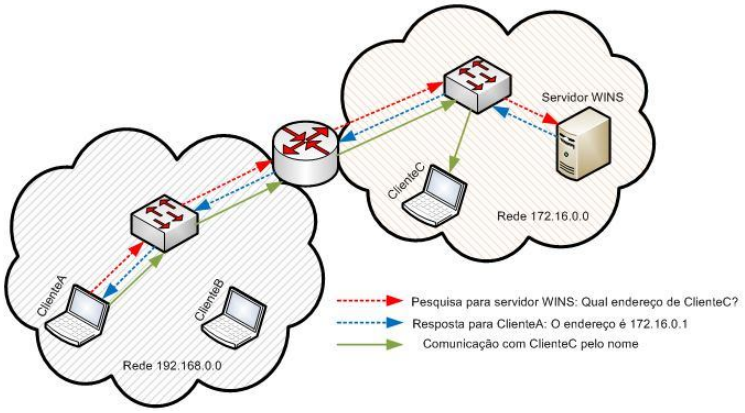

Figura 7 – Funcionamento do servidor WINS.
Fonte: (Romero, 2011).

- Lmhosts: Arquivo local encontrado geralmente no caminho *C:\System32\Drivers\Etc*. O arquivo consiste em um banco de dados estático que mapeia nomes NetBIOS específicos em endereço IP. Uma boa observação a ser feia em relação a esse tipo de consulta, é que ele só entrará em vigor no momento em que todos os métodos anteriores e DNS falharem. Nessa situação, devem-se adicionar manualmente as entrada.

3 MATERIAIS E MÉTODOS

3.1 MATERIAIS

Para criar o ambiente de trabalho, infraestrutura e realização dos testes deste trabalho utilizou-se o software de virtualização Oracle VM VirtualBox na versão 4.2.10 r84104.

Criou-se 3 máquinas virtuais. A Primeira com o Sistema Operacional Microsoft Windows Server Enterprise 2008 R2 para trabalhar com Serviço Active Directory e DNS. A segunda com Sistema Operacional Debian Squeeze 6.0 onde foi instalado e configurado o Samba. A terceira com Sistema Operacional Microsoft Windows XP Profissional para atuar como cliente.

O software VirtualBox foi escolhido por ser gratuito e de grande qualidade, abordando de forma simples a parte de instalação e configuração de máquinas virtuais para um trabalho rápido e fácil.

A versão do Oracle VM VirtualBox utilizada é suportada pela arquitetura 32 bits, mas independentemente as máquinas virtualizadas não precisam ser desta mesma arquitetura.

Para a comunicação entre as máquinas virtualizadas utiliza-se a rede interna do VirtualBox que por padrão utiliza a faixa de IP 192.168.56.0 mas pode ser configurada para outra facilmente.

3.2 DESCRIÇÃO DO WINDOWS UTILIZADO

Utilizada uma máquina cliente, com Microsoft Windows XP Profissional 32 bits para a realização de testes de inserção, autenticação, privilégios e remoção de usuários no domínio criado.

Para a simulação de um servidor Windows, utilizou-se a versão Microsoft Windows 2008 R2 Enterprise com linguagem PT_BR e arquitetura 32 bits.

Nesta máquina já instalado e configurado o Microsoft Active Directory.

3.2.1 INSTALAÇÃO E CONFIGURAÇÃO DO *ACTIVE DIRECTORY*

Para instalação deste serviço é necessário adicionar uma nova função ao servidor, como mostra a Figura 8.

Figura 8 - Adicionar função ao servidor Microsoft Windows.

Na janela do assistente para adição de funções, após avançar como mostra a Figura 9, assinala-se a opção "*Serviço de Domínio Active Directory*" que está presente na indicação de "*Funções*".

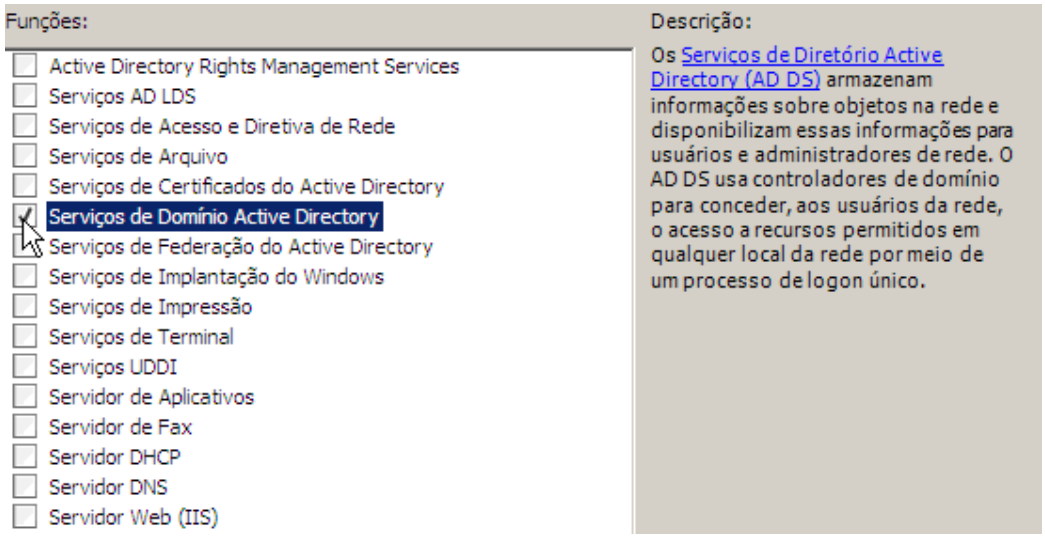

Figura 9 - Adicionando serviço de domínio Active Directory.

Ao lado dispõe de uma breve descrição do serviço selecionado. Com isso pode-se avançar para a próxima aba onde terão instruções e observações. Continuando uma aba para confirmação de instalação é exibida. Com a instalação já concluída, exibida na Figura 10 deve-se então executar o assistente de instalação do serviço, onde serão feitas as configurações.

Figura 10 - Assistente de instalação Microsoft Active Directory.

Com o assistente aberto escolhe-se o método de instalação em modo avançado antes de começar a configurar e se segue, uma mensagem com instruções úteis caso a rede possua máquina com Windows 98 ou Windows NT 4.0 para que possam acessar o ambiente do servidor, pode-se ir diretamente a aba seguinte onde é criado o domínio juntamente com a floresta. Para isto é necessário inserir o nome NetBIOS para o servidor de no máximo 15 (quinze) caracteres – será utilizado Windows – e outro nome no formato FQDN (Nome completo para o domínio) podendo ter 255 (duzentos e cinquenta e cinco) caracteres que será o DNS da rede interna – será usado windows.com.

Como está na Figura 11, o próximo passo é definir o nível da floresta com que o servidor irá trabalhar.

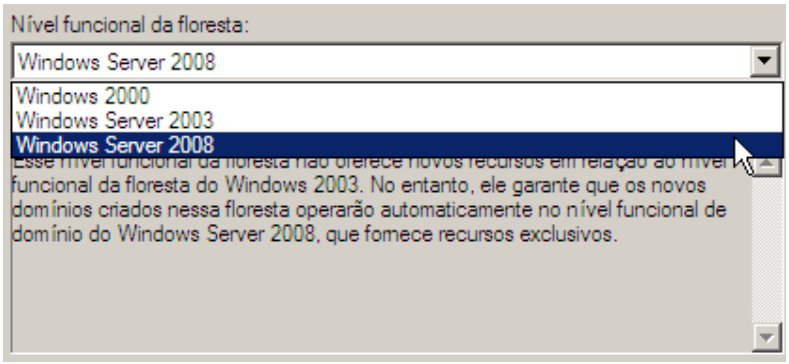

Figura 11 - Escolha de nível da floresta com que o Active Directory trabalhará.

Vale ressaltar que uma vez escolhida esta opção não pode ser alterada. Para efeito de teste será escolhida a opção “Windows Server 2008”.

Seguindo, será realizada uma verificação para ver se o DNS está configurado no servidor ou se a rede já possui outro em máquina diferente. Como

está sendo configurada a partir do zero, se faz necessária a criação do serviço. Para tal serviço é essencial que o servidor tenha IP estático. Depois da criação do DNS, como ilustra a Figura 12, o assistente exibe as pastas referentes ao banco de dados, pasta de log e SYSVOL (pasta utilizada para armazenar as políticas de grupos).

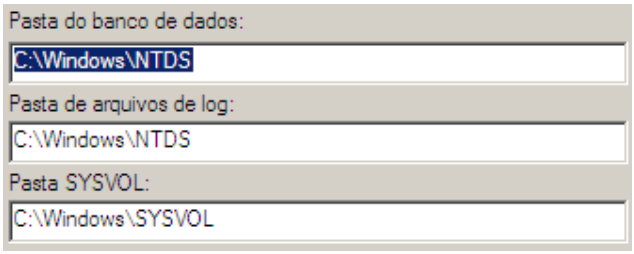

Figura 12 – Caminhos para as pastas com arquivos do Active Directory.

Na aba seguinte é preciso informar uma senha, utilizada para restauração da máquina, quando precisar iniciar o sistema em modo de segurança para fazer manutenção na base. Com isso passando a próxima etapa, o assistente mostra um resumo das configurações feitas podendo ser exportada no formato de texto e executada em outra instalação futuramente. Avançando na configuração o assistente começa a executar as configurações feitas e após terminar reinicia o servidor ativando o Active Directory junto com o DNS.

3.3 DESCRIÇÃO DO SAMBA UTILIZADO

Para a execução de servidor Linux, utiliza-se Debian Squeeze 6.0 sobre a plataforma 32 bits. Nesta máquina instala-se e se configura o Samba na versão 4.0.7.

O Samba nesta versão já possui complemento para integração de domínio para o Active Directory.

3.3.1 INSTALAÇÃO E CONFIGURAÇÃO DO SAMBA

Antes de obter o arquivo da Internet e começar a compilar e instalar é necessário instalar algumas dependências para que não ocorra nenhum problema com as etapas futuras. O Quadro 2 mostra a utilização do comando *apt-get install* seguido dos pacotes a serem instalados.

```
1 apt-get install build-essential libacl1-dev libattr1-dev libblkid-dev libgnutls-dev
2 libreadline-dev python-dev  python-dnspython gdb pkg-config libpopt-dev libldap2-dev
3 dnsutils libbsd-dev attr krb5-user acl docbook-xsl libcups2-dev libpam0g-dev
```

Quadro 2 - Requisitos básicos para instalação do Samba.
Fonte: SAMBA.ORG-1.

Para a próxima etapa da configuração é necessário a configuração do arquivo */etc/fstab* este trata as configurações do sistema. O Quadro 3 mostra as configurações que o arquivo deve conter.

```
1 /dev/hda3   /home   ext3     user_xattr,acl,barrier=1   1    1
```

Quadro 3 - Configuração do arquivo fstab.
Fonte: SAMBA.ORG-1.

Para ativar as configurações é preciso reiniciar a máquina. Contudo para verificar se as alterações foram realizadas com sucesso os comandos exibidos no Quadro 4 são indispensáveis.

```
1 touch test.txt2 setfattr -n user.test -v test
test.txt
3 setfattr -n security.test -v test2 test.txt
4 getfattr -d test.txt
5 file: test.txt
6 user.test="test"
7
8 getfattr -n security.test -d test.txt
9 file: test.txt
10 security.test="test2"
11
12 touch test3.txt
13 setfacl -m g:adm:rwx test3.txt
14 getfacl test3.txt
```

Quadro 4 - Teste para verificar a compatibilidade do sistema GNU/Linux .
Fonte: SAMBA.ORG-1.

As saídas obtidas nas *linhas 5 e 6* e nas *linhas 9 e 10* não retornam erros, portanto o sistema possui suporte completo para estes serviços. Para o comando

presente na linha 14 é preciso verificar se a saída possui em uma de suas linhas a mensagem *group:adm:rw.*

Tendo estas configurações prévias concluídas pode-se obter o pacote Samba 4.0.7 compactado e extraí-lo para uma pasta conforme o Quadro 5.

```
1 wget http://www.samba.org/samba/ftp/stable/samba-4.0.7.tar.gz
2 tar xfvz samba-4.0.7.tar.gz
```

Quadro 5 - Obtendo e extraindo o pacote Samba.

Através do comando presente na *linha 1* obtém-se o arquivo de um servidor FTP pertencente à organização responsável pelo site . A *linha 2* tem como finalidade descompactar o arquivo e pode receber parâmetros complementares para tal função.

Para a instalação pode ser visto no Quadro 6 os comandos necessários.

```
1 cd samba-4.0.7
2 ./configure --enable-debug --enable-selftest --prefix=/opt/samba
3 make
4 make install
```

Quadro 6 - Comandos de compilação e instalação Samba.
Fonte: SAMBA.ORG-2.

O comando utilizado na primeira linha tem a finalidade de trocar a pasta corrente. No Quadro 6 é alterado para o local ao qual se descompactou o arquivo Samba. A linha seguinte é utilizada para preparar o programa distribuído em código-fonte para ser compilado posteriormente com o comando da *linha 3* e instalado pelo comando presente na *linha 4*. O comando da *linha 2* pode receber parâmetros como complemento, da mesma forma que é exibido no Quadro 6. Para cada complemento há um significado importante e distinto, o complemento *--enable-debug* habilita o modo depurador o que facilita o diagnóstico em caso de erro, o parâmetro *--enable-selftest* mostra se o Samba pode se comportar de forma inesperada na plataforma utilizada e o parâmetro *--prefix* recebe em qual pasta será instalado.

4 RESULTADOS E DISCUSSÕES

4.1 INTEGRAÇÃO ACTIVE DIRECTORY E SAMBA

Para a integração entre Active Directory e Samba 4 uma detalhe de extrema importância é o fato de não ser necessário *provisionar*, ou seja, ativar o Samba como PDC se a intenção é integrá-lo.

Uma configuração básica entre servidores é alocar um endereço IP estático.

Para integração é necessário que a máquina Linux tenha em seu arquivo */etc/hostname* a adição do nome FQDN do Active Directory. No Quadro 7 é possível ver esta configuração.

```
1 linux.windows.com
```

Quadro 7 - Configuração do arquivo hostname.

Não é necessário a configuração de um servidor DNS para a máquina com sistema operacional Linux, a inserção das linhas contidas no Quando 8 e no Quadro 9 nos arquivos */etc/hosts e /etc/resolv.conf* respectivamente, é suficiente para o reconhecimento do servidor DNS configurado no servidor Windows.

```
1 192.168.56.110          linux.windows.com        linux
2 192.168.56.105          servidor.windows.com     windows
3 192.168.56.105          windows.com              dominio
```

Quadro 8 - Configuração do arquivo hosts.

```
1 nameserver    192.168.56.105
2 domain          windows.com
3 search          windows.com
```

Quadro 9 - Configuração do arquivo resolv.conf.

Com estas etapas o reconhecimento do servidor Windows já é realizado pelo servidor Linux.

Para que não haja problemas na configuração há a necessidade de excluir o arquivo *smb.conf* contido dentro da pasta */opt/samba/etc/* a qual foi especificado no comando ./configure com o parâmetro *--prefix*.

O próximo ajuste a ser realizado pertence ao protocolo Kerberos. Conforme o Quadro 10. No caminho /opt/samba/share/setup/ encontra-se o arquivo *krb5.conf* original, ou seja, com a estrutura padrão necessária para quem precisa configurá-lo, este arquivo será copiado através do comando presente na linha 1 para o local padrão utilizado para o leitura das configurações do protocolo Kerberos e as linhas seguintes indicam como deve ficar ajustado o arquivo com as mudanças finais.

```
1 cp /opt/samba/share/setup/krb5.conf /etc/krb5.conf
2
3 [libdefaults]
4 default_realm = WINDOWS.COM
5 dns_lookup_realm = true
6 dns_lookup_kdc = true
```

Quadro 10 - Configuração do arquivo krb5.conf.
Fonte: SAMBA.ORG-3

O domínio padrão representado pelo parâmetro *default_realm* deve ser escrito em letras maiúsculas, caso contrário não será reconhecido pelo protocolo. Para saber se as configurações foram feitas corretamente e o serviço está ativo, pode-se utilizar o comando *kinit* seguido do nome do usuário administrador do sistema para adquirir o ticket como mostra o Quadro 11.

```
1 kinit administrator
```

Quadro 11 - Adquirindo ticket do KDC.
Fonte: SAMBA.ORG-3.

Conforme exibido na Figura 13 através do comando *klist* é possível ver a validade do *ticket* adquirido pelo comando executado no Quadro 11, desta forma confirmando a comunicação entre os servidores.

```
root@linux:~# klist
Ticket cache: FILE:/tmp/krb5cc_0
Default principal: administrator@WINDOWS.COM

Valid starting     Expires            Service principal
08/07/13 00:02:50  08/07/13 10:03:17  krbtgt/WINDOWS.COM@WINDOWS.COM
        renew until 08/08/13 00:02:50
```

Figura 13 - Verificando ticket adquirido do KDC.

Com estas configurações realizadas e ativas pode-se integrar ambos os serviços para trabalho conjunto. Desta maneira o Quando 12 mostra o comando responsável por esta ação. Antes de ser executado é de grande importância parar o serviço através do comando *samba stop* caso o mesmo esteja ativo.

```
1 samba-tool domain join windows DC –Uadministrator --realm=servidor.windows.com
```

Quadro 12 - Comando para integração dos controladores de domínio.
Fonte: SAMBA.ORG-3.

Não havendo mensagens de erro significa que a integração e a replicação foram realizadas com sucesso e o servidor Linux estará presente no Active Directory como apresentado na Figura 14, podendo realizar configurações dentro de um grupo, criar ou alterar políticas de grupos, adicionar ou excluir usuário.

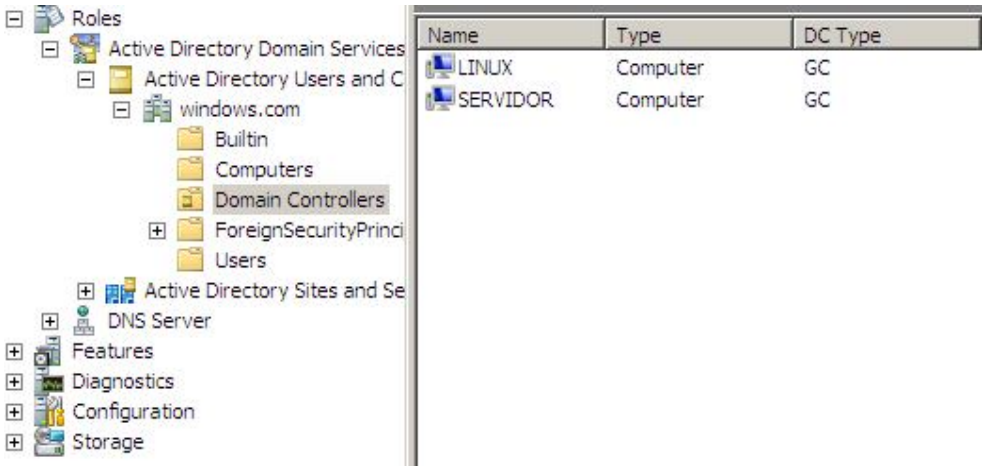

Figura 14 - Controladores presentes no Active Directory.

Contudo pode ocorrer um erro comum, mas pouco estudado, no momento de executar o comando do Quadro 12. Este problema, apresentado na Figura 15 ocorre devido um erro presente na replicação do SYSVOL.

```
Committing SAM database
descriptor_sd_propagation_recursive: DC=DomainDnsZones,DC=windows,DC=com not fou
nd under DC=windows,DC=com
descriptor_sd_propagation_recursive: DC=ForestDnsZones,DC=windows,DC=com not fou
nd under DC=windows,DC=com
Sending DsReplicateUpdateRefs for all the replicated partitions
Setting isSynchronized and dsServiceName
Setting up secrets database
Joined domain WINDOWS (SID S-1-5-21-316442391-1795407738-1419575065) as a DC
```

Figura 15 - Erro de replicação do SYSVOL.

Para correção é indispensável que sejam seguidos os comandos presentes no Quadro 13, assim o serviço de DNS passa a reconhecer o Samba 4 de forma apropriada.

```
1 /opt/samba/bin/samba-tool dns add 192.168.56.105 windows.com linux A
2 192.168.56.110 -Uadministrator
3/opt/samba/bin/ldbsearch -H /opt./samba/private/sam.ldb '(invocationid=*)' --cross-
4 ncs objectguid
5/opt/samba/bin/samba-tool dns add 192.168.56.105_msdcs.windows.com 71575df1-8624-
6 4003-801f-e4c81ba3a2c5 CNAME linux.windows.com -Uadministrator
```

Quadro 13 - Comandos para correção da integração entre os controladores de domínio.

Cada um dos comandos do Quadro 13 gera uma saída diferente. Para o comando das *linhas 1 e 2* o resultado da execução é exibido na Figura 16.

```
root@linux:~# /opt/samba/bin/samba-tool dns add 192.168.56.105 windows.com linux
 A 192.168.56.110 -Uadministrator
Password for [WINDOWS\administrator]:
Record added successfully
```

Figura 16 - Saída do primeiro comando presente no Quadro 13.

Para o comando das *linhas 3 e 4* o resultado da execução é mostrado na Figura 17.

```
root@linux:~# /opt/samba/bin/ldbsearch -H /opt/samba/private/sam.ldb '(invocatio
nid=*)' --cross-ncs objectguid
# record 1
dn: CN=NTDS Settings,CN=SERVIDOR,CN=Servers,CN=Default-First-Site-Name,CN=Sites,
CN=Configuration,DC=windows,DC=com
objectGUID: ef73a03b-8a18-4d6d-be19-c1d02e396e7b

# record 2
dn: CN=NTDS Settings,CN=LINUX,CN=Servers,CN=Default-First-Site-Name,CN=Sites,CN=
Configuration,DC=windows,DC=com
objectGUID: 71575df1-8624-4003-801f-e4c81ba3a2c5

# returned 2 records
# 2 entries
# 0 referrals
```

Figura 17 - Saída do segundo comando presente no Quadro 13.

Após o comando da Figura 17 ser executado é preciso encontrar o *record* que apresente o *CN* com o nome referente ao servidor Linux. Copiar o número apresentado pelo atributo *objectGUID* e executar o último comando do Quadro 13.

Este por sua vez terá uma saída diferente dos demais como pode ser visualizado na Figura 18.

Figura 18 - Saída do terceiro comando presente no Quadro 13.

Como forma de teste, para verificar se os comandos anteriores estão ativos e configurados de forma adequada executa-se os comandos mostrados no Quadro 14 que possuem como saída atributos do servidor Linux, como endereço IP e *hostname*.

```
1 host -t A linux.windows.com
2 host -t CNAME 71575df1-8624-4003-801f-e4c81ba3a2c5._msdcs.windows.com
```

Quadro 14 - Teste para verificar se o problema de replicação foi resolvido.

Neste momento pode-se validar o ticket adquirido pelo comando executado no Quadro 11, junto ao servidor Windows para que o acesso a base de dados do Active Directory seja liberada para o Samba através dos comandos apresentados no Quadro 15.

```
1 samba-tool drs kcc -Uadministrator servidor.windows.com
2 samba-tool drs showrepl
```

Quadro 15 - Validação do ticket adquirido.

Através do comando da *linha 1* é possível verificar a mensagem de sucesso, ao qual a autorização de acesso a base de dados foi concedida, apresentada na Figura 19. O comando presente na *linha 2* mostra a base de dados replicada, bem como os usuários presentes no Active Directory.

Figura 19 – Mensagem de liberação para acesso a base de dados .

A nova versão do Samba possui comandos nativos para administração de usuário e grupos. Com estes comandos é possível, por exemplo, adicionar, excluir e listar as informações contidas no PDC. Assim no Quadro 16 apresentam-se comandos básicos para que seja possível adicionar e listar respectivamente um usuário, deste modo podendo inserir uma máquina no domínio através deste usuário criado pelo Samba.

```
1 samba-tool user add testetl teste123@
2 samba-tool user list
```

Quadro 16 - Comandos para adicionar e listar usuários.

A senha precisa atender alguns requisitos de segurança, tais como:

- Possuir letra;
- Possuir número;
- Possuir caractere especial.

Para visualizar o usuário criado e verificar se a replicação esta funcionando perfeitamente atualiza-se o Active Directory e conforme a Figura 20 o usuário deve estar presente.

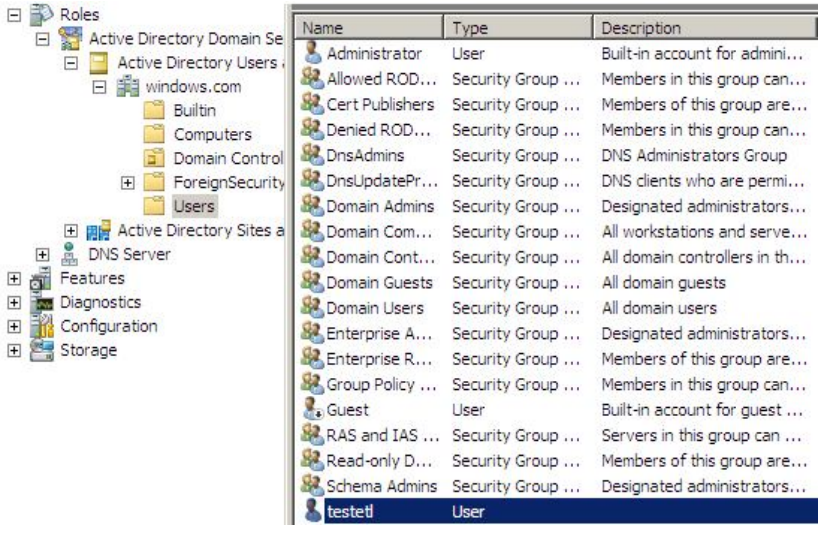

Figura 20 - Visualização do usuário criado pelo samba no Active Directory.

Com isto pode-se testar a máquina com Windows XP configurada de forma básica. Primeiramente é necessário adicionar a máquina no domínio. Isto é feito

acessando as propriedades do ícone *Meu Computador*, seguindo até a aba Nome do computador e acessando a janela aberta pelo botão *Alterar*. Seleciona-se o campo domínio e digite o nome NetBIOS configurado anteriormente para o Active Directory seguindo a ilustração da Figura 21. Informa-se *login* e senha do usuário administrador do domínio e confirma a ação.

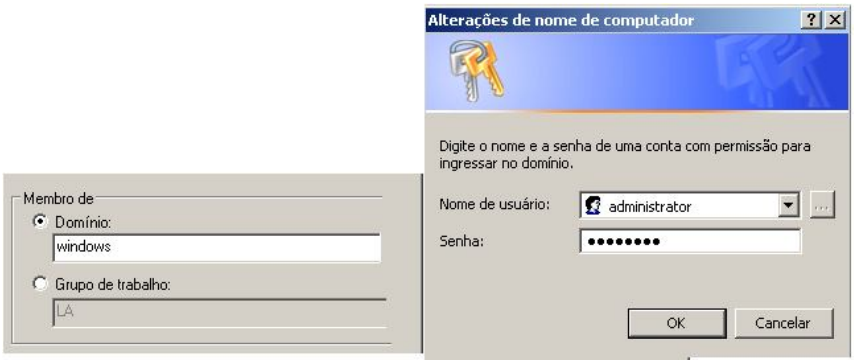

Figura 21 - Adicionando máquina ao domínio.

Com os dados digitados de forma correta, uma mensagem de boas vindas como a Figura 22 expõe será mostrada.

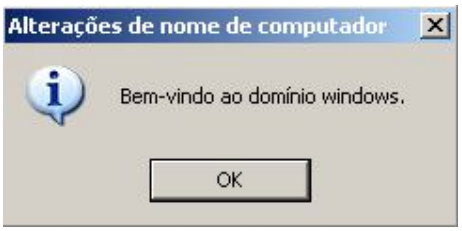

Figura 22 - Mensagem de boas vindas.

Para que as configurações entrem em vigor é necessário reiniciar a máquina. Por padrão é necessário utilizar a combinação de teclas *ctrl + alt + del* para ter acesso à tela de *logon* porém não mais é utilizado o usuário local para acessar o sistema operacional, desta forma fazendo uso do membro criado pelo PDC. Após digitar as informações do usuário e selecionar no campo fazer *logon* em o nome NetBIOS do servidor como ilustra a Figura 23 é possível acompanhar as configurações sendo carregadas. Estas informações podem ser, por exemplo,

políticas de grupos, permissões de acesso e unidades mapeadas configuradas no Active Directory ou a partir deste momento no próprio Samba.

Figura 23 - Logon no Windows

5 CONSIDERAÇÕES FINAIS

5.1 CONCLUSÃO

A instalação do Samba 4 precisa ocorrer de maneira que atenda as configurações iniciais e aos requisitos básicos corretamente, para que não haja problemas futuros de funcionamento. A integração por sua vez necessita ser feita com extrema atenção nas configurações, uma vez que é possível receber uma mensagem informando que a integração foi realizada corretamente, enquanto podem ter ocorrido problemas na replicação da base de dados entre os controladores sem que se tenha percebido. O administrador deve ficar atento a cada etapa concluída e verificar a saída de cada comando executado, analisando se está de acordo com o esperado. Com esta integração fica a critério do administrador utilizar o controlador que melhor satisfaz sua necessidade no momento.

A utilização do Samba 4 como um controlador de domínio, integrado com o Active Directory, podendo efetuar configurações e manutenção na base de dados que será replicada automaticamente, provê uma qualidade do serviço maior, juntamente com uma maior confiabilidade na segurança das informações, fornecendo maior flexibilidade nas configurações e também reduzindo o custo com aquisições feitas para a infraestrutura da rede, uma vez que o controlador Samba pode assumir as responsabilidades de controlador primário quando o Active Directory não estiver disponível ou ainda responder as requisições feitas pelos clientes quando o Active Directory estiver sobrecarregado.

Foram necessários diversos testes de configuração e integração, sendo o erro citado no capítulo 4 o mais provável e comum de acontecer. Após a resolução deste problema de replicação da base de dados é possível gerenciar ambos de maneira simples. Usuários foram adicionados, removidos e alterados e como esperado a base foi replicada. Clientes foram ingressados no domínio e foi possível o acesso a máquina através destes usuários.

Analisando os pontos citados pode-se concluir que a integração do Samba 4 com Active Directory dentro do ambiente proposto satisfez completamente as expectativas e objetivos do trabalho, mesmo considerando que a versão utilizada

neste trabalho ainda esteja em desenvolvimento no momento da realização deste trabalho.

5.2 TRABALHOS FUTUROS

Para complementar este trabalho pode se estudar as seguintes configurações dentro de uma rede que utilize os controladores Active Directory e Samba 4 com a integração realizada:

- Gerenciamento de usuários;
- Gerenciamento de grupos;
- Aplicar política de grupos;
- Compartilhamento de arquivos;
- Compartilhamento de dispositivos.

6 REFERÊNCIAS BIBLIOGRÁFICAS

CONECTIVA. **Kerberos. Autenticação do Sistema**. Disponível em: <http://www.conectiva.com/doc/livros/online/10.0/servidor/pt_BR/ch13s04.html>. Acesso em: 16 de Fevereiro de 2013.

COSTA, P. H. A. **Samba: Windows e Linux em rede**. São Paulo: Linux New Media do Brasil, 2010.

ERICH. S. M. **Autenticação Integrada Baseada em Serviço de Diretório LDAP**. Disponível em: <http://www.linux.ime.usp.br/~cef/mac499-06/monografias/erich/html/ch01s05.html>. Acesso em: 18 de Fevereiro de 2013.

ERICOM. “Kerberos in PowerTerm Solutions”. Disponível em: <http://www.ericom.com/kerberos.asp>. Acesso em: 17 de Abril de 2013.

FILHO.M.M.C. **Kerberos. Apresentação do protocolo Kerberos**. Disponível em: <http://www.gta.ufrj.br/grad/99_2/marcos/kerberos.htm>. Acesso em: 20 de Fevereiro de 2013.

LEVIS. “DHCP Pictures Photo Gallery added by Levis”. Disponível em: <http://withfriendship.com/user/levis/dhcp.php>. Acesso em: 10 de Fevereiro de 2013.

MICROSOFT. **Simple Network Time Protocol**. Disponível em: <http://msdn.microsoft.com/pt-br/library/aa919019.aspx>. Acesso em: 10 de Março de 2013a.

MICROSOFT. **Protocols and Interfaces to Active Directory**. Disponível em: <http://technet.microsoft.com/pt-br/library/cc961766%28en-us%29.aspx>. Acesso em: 10 de Março de 2013b.

MICROSOFT. **Definição de WINS**. Disponível em: <http://technet.microsoft.com/pt-br/library/cc784707%28v=ws.10%29.aspx>. Acesso em: 18 de Maio de 2013c.

MINASI. M; ANDERSON. C.; SMITH. B.M; TOOMBS. D. **Dominando o Windows 2000 Server**. *São Paulo: Pearson Education do Brasil*, 2001. 1275 p.

NTP. “The Network Time Protocol”. Disponível em: <http://ntp.org/>. Acesso em: 10 de Abril de 2013.

ROMERO. **NetBIOS, como funciona?**. Disponível em: <http://blog.romerojunior.com/microsoft/netbios-como-funciona/>. Acesso em: 18 de Maio de 2013.

SAMBA.ORG. **Samba4/ OS Requeriments**. Disponível em: <http://wiki.samba.org/index.php/Samba_4_OS_Requirements>. Acesso em: Fevereiro de 2013a.

SAMBA.ORG. **Samba AD DC HOWTO**. Disponível em: <http://wiki.samba.org/index.php/Samba4/HOWTO#Step_1:_download_Samba4>. Acesso em: Fevereiro de 2013b.

SAMBA.ORG: **Samba4/HOWTO/Join a domain as a DC**. Disponível em: <http://wiki.samba.org/index.php/Samba4/HOWTO/Join_a_domain_as_a_DC>. Acesso em: 13 de Julho de 2013c.

SCRIMGER.R.; LASALLE.P.; PARIHAR.M.; GUPTA.M. **TCP/IP - A Bíblia**. Rio de Janeiro: Campus, 2002. p.642.

SHARPE, R. **Just what is SMB?**. Disponível em: <http://www.samba.org/cifs/docs/what-is-smb.html>. Acesso em: 20 de Maio de 2013.

THE OPENLDAP FOUNDATION. **OpenLDAP 2.1 Administrator's Guide**. Disponível em: <http://www.bind9.net/manual/openldap/2.1/index.html>. Acesso em: 20 de Março de 2013.

TRIGO.C.H. **OpenLDAP - Uma Abordagem Integrada**. São Paulo: Novatec, 2007. p.239.

UFRJ. **Apresenta o funcionamento do protocolo CIFS**. Disponível em: <http://www.gta.ufrj.br/grad/01_2/samba/smbcifsinternamente.htm>. Acesso em: 20 de Março de 2013.

UFRGS. **NetBEUI/NetBIOS**. Disponível em: <http://penta.ufrgs.br/redes296/cliente_ser/redes_.htm#netbios>. Acesso em: 03 de Abril de 2013.

WINCO. **O que é DNS?**. Disponível em: <http://ddns.winco.com.br/dns/>. Acesso em: 20 de Maio de 2013.