

ALESSANDRO ERIK DE JESUS

**IMPLANTAÇÃO DE AUTENTICAÇÃO DE SERVIÇOS LINUX NO SERVIÇO DE
DIRETÓRIO ACTIVE DIRECTORY**

**Monografia apresentada ao curso de
Especialização em Redes de
Computadores da Faculdade
Natalense para o Desenvolvimento do
Rio Grande do Norte como requisito
obrigatório para a obtenção do título
de Especialista em Redes de
Computadores.**

**Orientador: Aluizio Ferreira da Rocha
Neto (M.Sc)**

**NATAL/RN
2010**

LIGA DE ENSINO DO RIO GRANDE DO NORTE
FACULDADE NATALENSE PARA O DESENVOLVIMENTO DO RN
ESPECIALIZAÇÃO EM REDES DE COMPUTADORES

Alessandro Erik de Jesus

IMPLANTAÇÃO DE AUTENTICAÇÃO DE SERVIÇOS LINUX UTILIZANDO
SERVIÇO DE DIRETÓRIO ACTIVE DIRECTORY

Monografia apresentada a Faculdade Natalense para o Desenvolvimento do Rio Grande do Norte e aprovada em ____ de _____ de _____, pela banca examinadora composta pelos seguintes membros

Aluizio Ferreira da Rocha Neto, M.Sc.
Orientador

Membro examinador

Membro examinador

RESUMO

Esse trabalho faz um estudo sobre a autenticação de usuários Windows cadastrados no Active Directory em serviços Linux, apresentando como modelo a autenticação desses usuários em um serviço de Proxy Squid, mostrando como adicionar ou remover acessos por grupos de usuários. A finalidade é ter uma granularidade maior de restrições e bloqueios além de ter uma forma mais precisa identificar os acessos feitos pelos usuários.

Palavras-chave: Active Directory. NTLM. Linux. Proxy. Windows. Squid. Integração Windows e Linux.

ABSTRACT

This present study is about authentication of users registered in Windows Active Directory for services on Linux, using authentication of users on a Squid Proxy service as model, showing how to add or remove access by user groups. The aim is to have a larger granularity of restrictions and blockages in addition to having a more precise way to identify accesses made by users.

Keywords: Active Directory. NTLM. Linux, Proxy. Windows. Squid. Linux and Windows Integration.

SUMÁRIO

1 INTRODUÇÃO	8
1.1 PROBLEMA DE PESQUISA	8
1.2 OBJETIVOS	9
1.2.1 Objetivos Específicos	9
1.3 JUSTIFICATIVA	9
2 REFERENCIAL TEORICO	10
2.1 AUTENTICAÇÃO DE USUÁRIOS	10
2.2 PROTOCOLOS DE AUTENTICAÇÃO	10
2.2.1 ntlm	10
2.2.2 Kerberos	12
2.3 ACTIVE DIRECTORY	14
2.3.1 Componentes da Estrutura Lógica	15
2.3.2 Estrutura Física	18
2.4 FERRAMENTAS LINUX	19
2.4.1 squid	19
2.4.2 dansguardian	21
3 DESENVOLVIMENTO DA SOLUÇÃO.....	22
3.1 CONFIGURAÇÃO DO ACTIVE DIRECTORY	22
3.2 CONFIGURAÇÃO DO SERVIDOR PROXY	23
3.3 GERANDO RELATÓRIOS DE ACESSOS USANDO O SARG.....	31
3.4 DANSGUARDIAN	33
4 CONCLUSÃO	39
REFERÊNCIAS	40

LISTA DE FIGURAS

Figura 1 - Protocolo de Autenticação NTLM.....	11
Figura 2 - Autenticação Kerberos.....	13
Figura 3 - Estrutura Lógica do Active Directory.....	14
Figura 4 - Serviço de Diretório.....	15
Figura 5 - Unidades Organizacionais no AD.....	16
Figura 6 – Domínio Fábrica	17
Figura 7 - Estrutura Física do Active Directory	18
Figura 8 - Proxy Squid	20
Figura 9 - Dansguardian como Intermediário entre o Proxy e o Usuário.....	21
Figura 10 - Configuração do Active Directory.....	22
Figura 11 – Estrutura de Usuários e Grupos no AD.....	23
Figura 12 – Pacotes do Servidor Proxy	23
Figura 13 – Configuração para o servidor Kerberos	24
Figura 14- Configuração do arquivo /etc/squid/squid.conf	25
Figura 15 - Adicionando Máquina linux no Domínio Windows.....	26
Figura 16 - Servidor Proxy cadastrado no AD	27
Figura 17 - Editando o arquivo /etc/nsswitch.conf.....	28
Figura 18 - Usando o wbinfo para listar os usuários do AD	28
Figura 19 - Usando o comando getent.....	29
Figura 20 - Configuração /etc/squid.conf	29
Figura 21- Criando acls para grupos no Windows	30
Figura 22 - Configurando o proxy no cliente	31
Figura 23 - Configuração do sarg.conf.....	32
Figura 24 - Relatório usando SARG	33
Figura 25 - Baixando o Pacote do Dansguardian	34
Figura 26 - Configurando arquivos para compilação	35
Figura 27 - Alterando a permissão da pasta de log	35
Figura 28 - Arquivo /etc/init.d/dansguardian	36
Figura 29 - Adicionando o Daemon na inicialização do sistema	36
Figura 30 - Configuração do arquivo /etc/dansguardian.conf - Parte 1	37
Figura 31 - Configuração do arquivo /etc/dansguardian.conf - Parte 2	37
Figura 32 - Alteração do navegador pelo usuário.....	38

LISTA DE ABREVIATURAS

AD	- Active Directory
RADIUS	- Remote Authentication Dial In User Service
NTLM	- NT LAN Manager
HTTP	- Hypertext Transfer Protocol
HTTPS	- Hypertext Transfer Protocol Secure
MIT	- Massachusetts Institute of Technology
DES	- Data Encryption Standard
DNS	- Domain Name System
NCSA	- National Center for Supercomputing Applications
LDAP	- Lightweight Directory Access Protocol
SMB	- Server Message Block
SARG	- Squid Analysis Report Generator

1 INTRODUÇÃO

Muitas organizações têm de mais valioso em seu patrimônio a informação que possuem, ela é muito mais valiosa do que seus patrimônios físicos como prédios, máquinas e outros bens. A necessidade de organizar, gerenciar e compartilhar de melhor forma essas informações fez com que se desenvolvessem os sistemas computacionais e aperfeiçoassem as telecomunicações.

Um dos desafios dos administradores de redes é a administração de usuários e dos recursos computacionais de forma centralizada, pois a grande heterogeneidade dos sistemas operacionais (Linux, Windows, Unix, etc.) e de tecnologias corporativas (Mysql, SQLServer, Servidores RADIUS, de E-mails, etc.) em um mesmo ambiente corporativo gera um problema de duplicidade de credenciais. Pode-se ter uma credencial para o computador de trabalho, uma para acessar a conta de e-mail, outro para acessar o banco de dados e assim em diante.

Para resolver esse problema o administrador pode usar serviços de diretório onde podemos localizar as credencias do usuário pela rede utilizando um servidor que armazena essas informações. Em ambientes que usam o Windows como sistema operacional de seu parque computacional é comum encontrar um servidor Windows com o Active Directory(AD), utilizando-o como um “autenticador” em um ambiente de rede.

Porém, alguns dos serviços que não são ambientados no Windows podem fazer a autenticação dos seus usuários em um servidor que tem o AD, desde que usem bibliotecas que façam a comunicação com o servidor, um exemplo disso é o servidor de proxy Squid que pode autenticar seus usuários utilizando o protocolo NTLM da Microsoft.

1.1 PROBLEMA DE PESQUISA

O problema de pesquisa abordado por este trabalho foi estudar como fazer a autenticação de usuários que estão armazenados no Active Directory em serviços Linux, em especial em um Proxy Squid com filtro de conteúdo Dansguardian, usando as credenciais de usuário que são utilizadas para acessar as máquinas Windows que estão em um domínio através do protocolo NTLM. Isso se faz necessário porque quando se usa um proxy transparente o administrador não sabe qual o usuário está acessando o site visitado além de ele não identificar acessos a sites que usam HTTPS. Identificando para cada usuário o site ao qual ele conectou.

1.2 OBJETIVOS

O objetivo desse projeto é mostrar como fazer a autenticação de usuários de serviços Linux no Active Directory, mostrando mais ênfase em um servidor *proxy* Squid e como configurar clientes Windows para utilizar esse *proxy*

1.2.1 Objetivos Específicos

Para atingir o objetivo geral, as seguintes etapas serão seguidas:

- Implantar uma rede com domínio Windows registrado num AD
- Mostrar como integrar uma máquina Linux para ser adicionada num domínio
- Configurar o servidor Linux para aceitar a autenticação dos seus serviços no AD

1.3 JUSTIFICATIVA

A motivação para esse trabalho foi para apresentar como as novas tecnologias integradas ao ambiente corporativo podem apresentar melhorias na qualidade e na gestão dos recursos de rede, pois sabemos que esses recursos precisam ser gerenciados de forma otimizada para aumentar a produtividade e a qualidade dos serviços que estão implantados na organização.

Além disso, esse trabalho pretende mostrar como ambientes Windows e Linux possuem serviços que podem ser integrados, inspirando estudantes e profissionais de redes a utilizarem esses serviços em seus projetos.

2 REFERENCIAL TEORICO

2.1 AUTENTICAÇÃO DE USUÁRIOS

A autenticação de usuários em uma rede é quando um usuário pretende acessar um sistema ou uma rede utilizando para isso um nome de usuário (login) e uma senha, que é verificada pelo banco de dados do sistema. Em caso de identidade confirmada, o sistema libera o perfil do usuário de acordo com as suas permissões.

2.2 PROTOCOLOS DE AUTENTICAÇÃO

2.2.1 ntlm

NTLM é um protocolo do método desafio-resposta entre um cliente (que deseja se autenticar) e um servidor (onde o cliente se autentica). A negociação (handshake) é feita com três mensagens: o tipo 1(negociação), tipo 2(desafio) e tipo 3(authenticação). Cada mensagem tem em seu cabeçalho a string “NTLMSSP”.

O protocolo funciona da seguinte forma:

1. O cliente manda uma mensagem GET para o servidor ou o proxy, o servidor responde com uma mensagem de status 401 e solicita a autenticação NTLM
2. Agora, o cliente envia a mensagem de tipo 1, com as informações necessárias (nome de usuário do Windows, o nome do domínio, o hash da senha, o tamanho da chave de criptografia, e o pedido de autenticação compartilhada) para o servidor. O servidor responde com uma mensagem de tipo 2 e mensagem de status 401. Além disso, ele envia uma chave aleatória para que o cliente possa se autenticar.
3. Na terceira mensagem o cliente usa a informação recebida pelo servidor mais os dados de autorização do Windows, envia os dados para o servidor (tipo 3) e se tudo estiver OK o servidor libera o usuário.

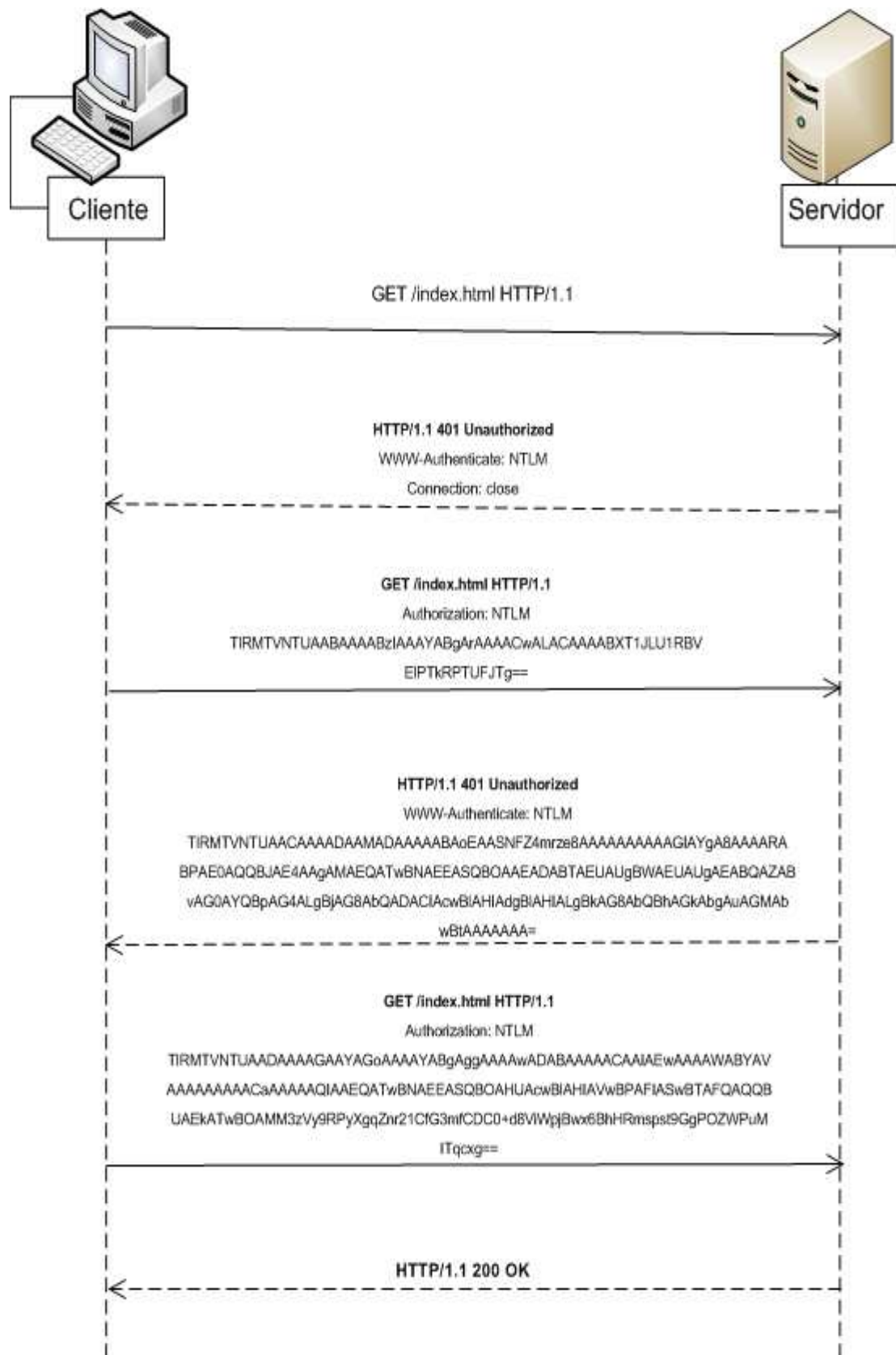


Figura 1 - Protocolo de Autenticação NTLM
Fonte: Próprio Autor

2.2.2 Kerberos

É um protocolo de autenticação desenvolvido pelo MIT e provê uma autenticação cliente/servidor, para segurança ele usa criptografia simétrica, ex: DES.

De acordo com (CALÔR FILHO, 2009), No Kerberos, o usuário apresenta um *ticket*, fornecido pelo *Kerberos authenticator server* (AS), assim como a carteira de motorista é fornecida pelo DETRAN. O serviço então examina o *ticket* para verificar a identidade do usuário. Se tudo estiver ok o usuário é aceito.

Para esta explicação, imaginemos que as mensagens são escritas em papel, e são criptografadas colocando-as em uma caixa-forte, associada a uma chave.

1. Primeiro o usuário envia uma mensagem ao AS: "Eu, J. Random User, gostaria de falar com o servidor Foo".
2. Quando o AS recebe a mensagem, ele faz duas cópias de uma nova chave registrada. Estas chaves são chamadas de chaves de sessão. Elas serão usadas nas trocas diretas entre serviço e usuário.
3. Ele coloca uma das chaves de sessão na Caixa 1, junto com um pedaço de papel com o nome "Servidor Foo" escrito. A caixa é trancada com a chave do usuário (o AS conhece todas as chaves de usuário e todas as chaves de serviço).
4. Por que este papel aqui? Devemos nos lembrar que a caixa é na realidade apenas uma mensagem criptografada, e que a chave de sessão é uma sequência randômica de bytes. Se a Caixa 1 somente contivesse a chave de sessão, o usuário não teria como reconhecer se a resposta veio do AS, ou não saberia se a decifração teve sucesso. Pela adição da mensagem "Servidor Foo", o usuário (mais precisamente a aplicação do usuário) poderia saber se a caixa veio do AS, e se a decifração obteve sucesso.
5. Ele coloca a outra chave de sessão em uma Caixa 2, junto com um pedaço de papel com o nome "J. Random User". Esta caixa é fechada com a chave do serviço.
6. Ele envia ambas as caixas ao usuário.
7. Na versão 4 do Kerberos, a Caixa 2 era colocada (sem necessidade) dentro da caixa 1, mas na versão 5 isto foi corrigido.
8. O usuário destranca a Caixa 1 com sua chave, extraindo assim a chave de sessão e o papel com o nome "Servidor Foo" escrito nele.

9. O usuário não consegue abrir a Caixa 2 (ela foi trancada com a chave do serviço, que só o AS e o serviço conhecem). Então ele coloca um pedaço de papel com a hora corrente numa Caixa 3, e tranca esta com chave de sessão, e envia ambas ao serviço.
10. O serviço abre a Caixa 2 com sua própria chave, extraindo a chave de sessão e o papel com "J. Random User" escrito nele. Ele abre a Caixa 3 com a chave de sessão para extrair o pedaço de papel com a hora corrente nele. Estes itens demonstram a identidade do usuário.

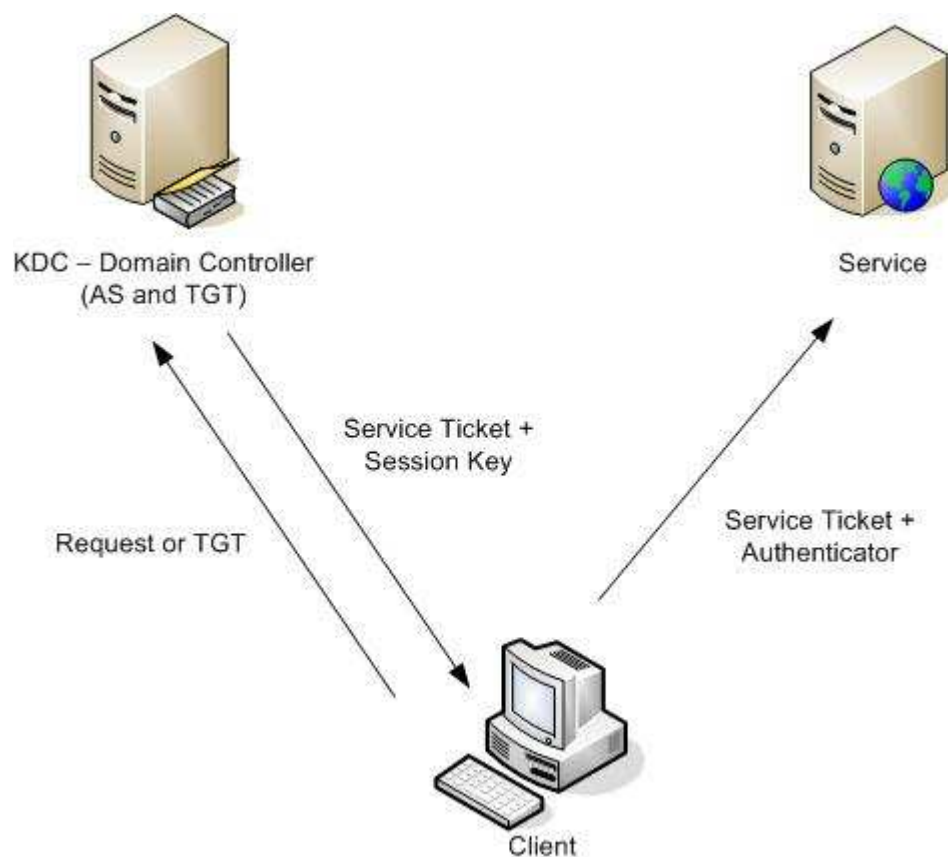


Figura 2 - Autenticação Kerberos
Fonte: adopenstatic.com

2.3 ACTIVE DIRECTORY

Devido a necessidade de criar um serviço para centralizar os usuários e recursos, foi desenvolvido para plataformas Windows Server o “Active Directory”.

Em uma rede do Microsoft Windows Server, o serviço de diretório Active Directory proporciona a estrutura e as funções para organizar, administrar e controlar o acesso aos recursos de rede. O Active Directory também permite administrar de forma central a rede do Windows Server. Esse recurso significa que é possível armazenar de forma central informações sobre a empresa, por exemplo, informações de usuários, grupos e impressoras, e que os administradores podem administrar a rede de um único lugar. (CORPORATION, 2003, p.1)

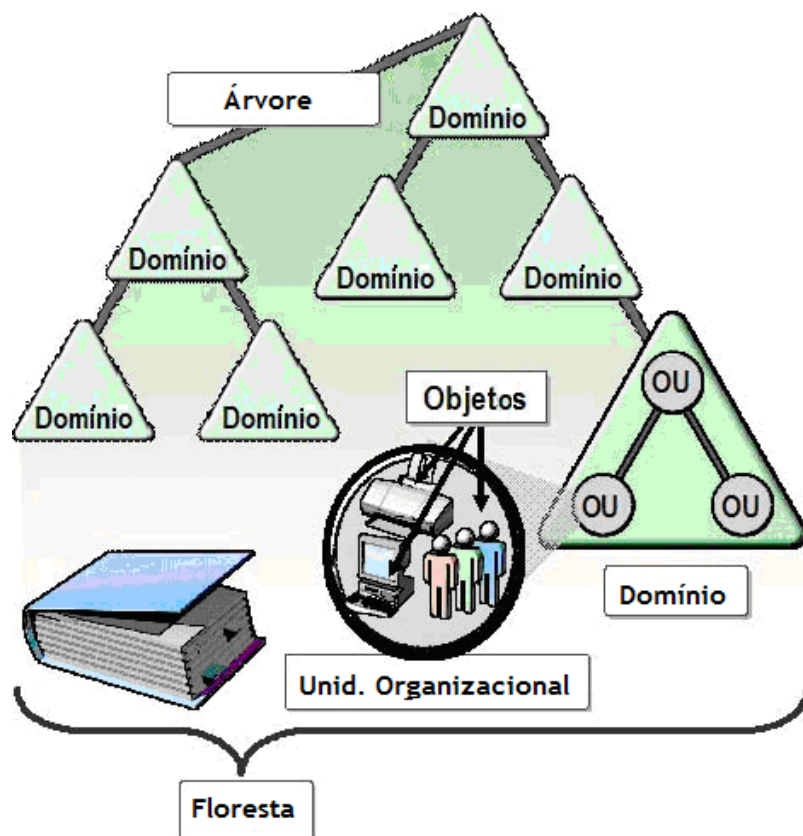


Figura 3 - Estrutura Lógica do Active Directory
Fonte: CORPORATION (2003, p. 2)

O Active Directory é o serviço de diretório de uma rede do Windows Server. Um serviço de diretório armazena informações sobre os recursos da rede e permite que os mesmos estejam acessíveis aos usuários e aos aplicativos. Os serviços de diretório proporcionam uma forma coerente de nomear, descrever, localizar, obter acesso, administrar e proteger as informações relativas aos recursos da rede.

(CORPORATION,2003, p. 2)

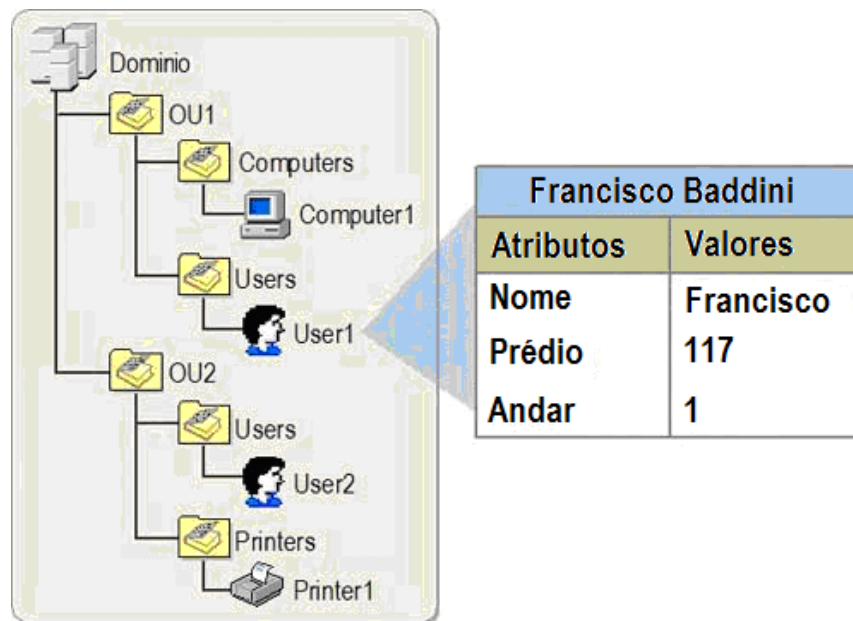


Figura 4 - Serviço de Diretório
Fonte: CORPORATION (2003, p. 7)

2.3.1 Componentes da Estrutura Lógica

- o **Objetos**. Eles são os componentes básicos da estrutura lógica.
- o **Classes de objeto**. São os modelos de tipos de objetos que podem ser criados no Active Directory. Cada classe de objeto é definida por um grupo de atributos que estabelece os valores que podem ser associados a um objeto. Cada objeto tem uma combinação única de valores de atributos.
- o **Unidades Organizacionais**: São contêineres de Objeto com características semelhantes ou que seja parte de uma outra Unidade Organizacional. Pode-se fazer delegações de Unidades Organizacionais.

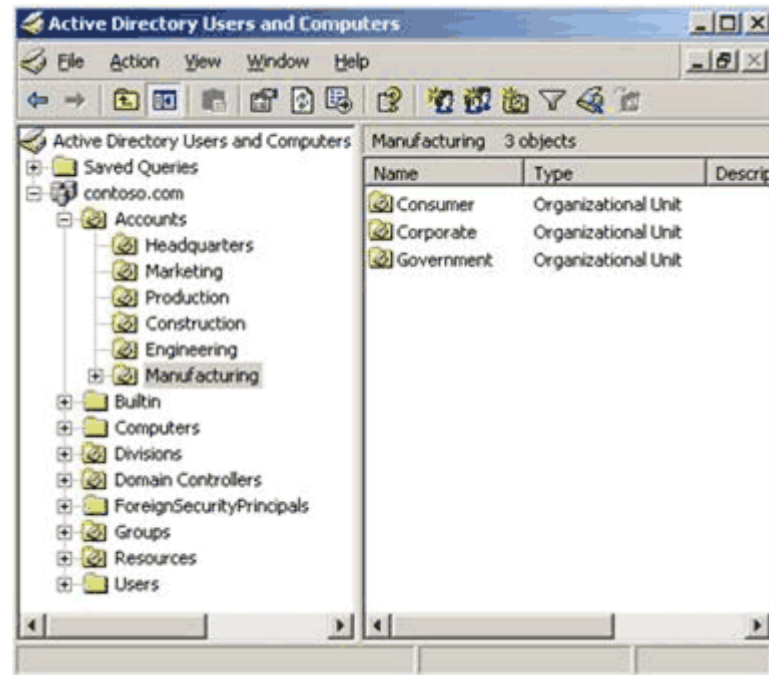


Figura 5 - Unidades Organizacionais no AD
CORPORATION (2003, p. 7)

o **Domínios**. São as unidades funcionais básicas da estrutura lógica do Active Directory e, portanto, é uma coleção de objetos administrativos definidos que compartilham, através de um banco de dados comum do diretório, diretivas de segurança e relações de confiança com outros Domínios. Os domínios oferecem as 3 funções a seguir:

- Um limite administrativo para os objetos
- Meios de administrar a segurança dos recursos compartilhados
- Uma unidade de replicação para os objetos

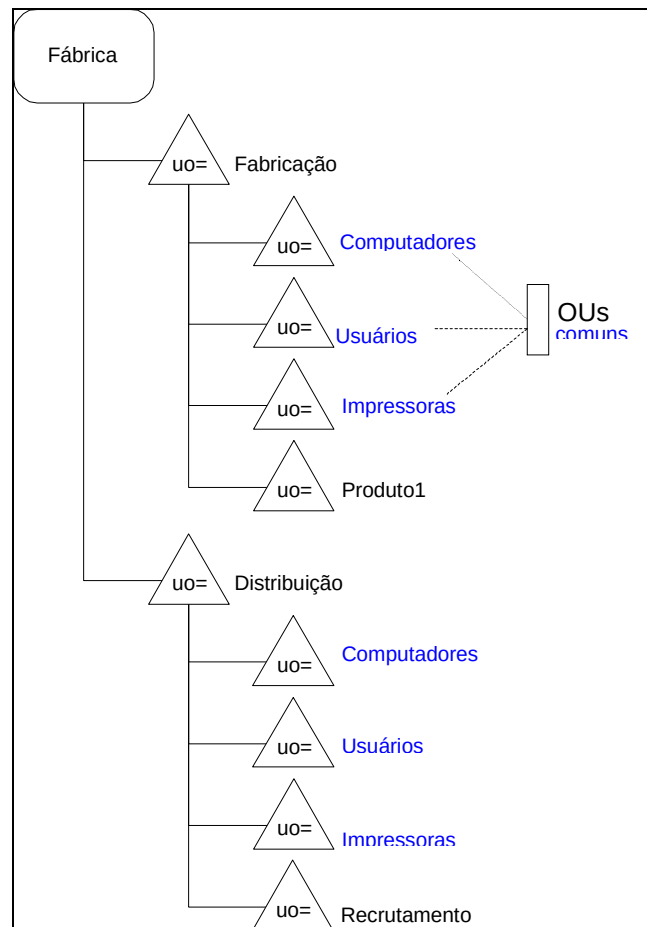


Figura 6 – Domínio Fábrica
Fonte: Notas de Aula

- o **Árvores de domínio.** São Domínios agrupados em estruturas hierárquicas. Quando um segundo domínio é adicionado a uma árvore, ele é convertido em Filho da árvore Raiz do Domínio. O domínio ao qual um Filho do Domínio é adicionado é chamado de Domínio Pai. O Domínio Filho pode ter seus próprios Domínios Filhos e seu nome é combinado com o nome do seu Domínio Pai para formar o seu próprio nome exclusivo, o DNS (Domain Name System). Um exemplo seria corp.nwtraders.msft. Desse modo, uma árvore tem um Nome de Espaço contínuo.
- o **Florestas.** Uma Floresta é uma instância completa do Active Directory, e consiste em uma ou mais árvores. Em uma única árvore de 2 níveis, recomendável para a maioria das organizações, todos os Domínios Filhos são filhos do Domínio Raiz da Floresta para formar uma árvore contígua. O primeiro domínio na floresta é chamado de Domínio Raiz da Floresta e o nome

desse domínio faz referência à floresta, por exemplo, nwtraders.msft. Por padrão, as informações no Active Directory só são compartilhadas dentro da floresta. Dessa forma, a segurança da floresta estará contida em uma única instância do Active Directory.

2.3.2 Estrutura Física

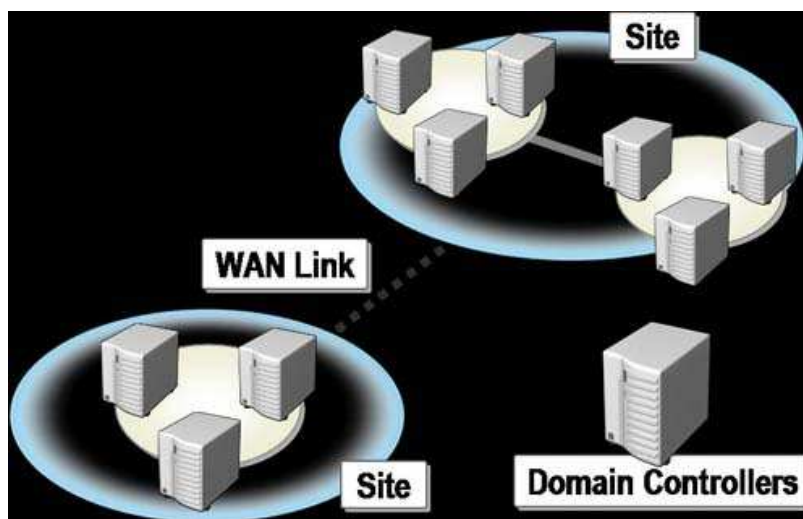


Figura 7 - Estrutura Física do Active Directory
Fonte: CORPORATION (2003, p. 3)

Em comparação com a estrutura lógica e os requisitos administrativos dos modelos, a estrutura física do Active Directory otimiza o tráfego da rede, determinando como e quando ocorre a replicação e o tráfego do logon. Para otimizar o uso da largura de banda da rede Active Directory, você precisa entender a sua estrutura física.

Os elementos da estrutura física do Active Directory são:

- o **Controladores de domínio.** Estes computadores executam o Microsoft® Windows® Server 2003 ou o Windows 2000 Server e o Active Directory. Cada Controlador de Domínio realiza funções de armazenamento e replicação e, além disso, oferece suporte a apenas um domínio. Para garantir uma disponibilidade contínua do Active Directory, cada domínio deve ter mais de um controlador de domínio.
- o **Sites do Active Directory** Os sites são grupos de computadores conectados. Quando você estabelece sites, os Controladores de Domínios que estão dentro de um mesmo site podem se comunicar com frequência. Essa comunicação reduz ao mínimo o estado de latência dentro do site, isso é, o tempo necessário para que

uma modificação realizada em um Controlador de Domínio seja duplicada nos outros controladores de domínio. Você cria sites para otimizar o uso da largura de banda entre controladores de domínio em diversos locais.

- o **Partições do Active Directory** Cada Controlador de Domínio contém as seguintes partições do Active Directory:
 - **Partições de Domínio**, que contêm a replicação de todos os objetos neste domínio. Essa partição é duplicada apenas para outros Controladores de Domínio do mesmo domínio.
 - **Partição de Configuração**, que contém a topologia da floresta. A topologia registra todas as conexões dos Controladores de Domínio na mesma floresta.
 - **Partição de Esquema**, que contém o esquema da floresta. Cada floresta tem um esquema de modo que a definição de cada classe do objeto seja constante. As partições de Configuração e Esquema de Partições são duplicadas para cada Controlador de Domínio na floresta.

Opções de Partição de Aplicativos que contêm os objetos relacionados à segurança e são utilizados por um ou mais aplicativos. As partições de aplicativos são duplicadas em Controladores de Domínio específicos na floresta.

2.4 FERRAMENTAS LINUX

2.4.1 squid

Segundo (WESSELS, 2004), trata-se de uma ferramenta capaz de aceitar requisições HTTP e HTTPS de clientes e capaz de efetuar requisições HTTP, FTP e Gopher para servidores, além de implementar várias características úteis em ambientes corporativos:

- Controle de banda no acesso a Internet;
- Redução do tempo de carregamento de páginas na Internet;
- Coleta de estatísticas do tráfego de acesso a Internet proveniente da rede local;
- Bloqueio de sites considerados de conteúdo inadequado;
- Garantia de que somente os usuários autorizados terão acesso a Internet;
- Conversão de requisições HTTPS de um lado em HTTP do outro lado;
- Proteção de máquinas internas de acessos externos uma vez que as requisições a sites externos são efetuadas pelo Proxy.

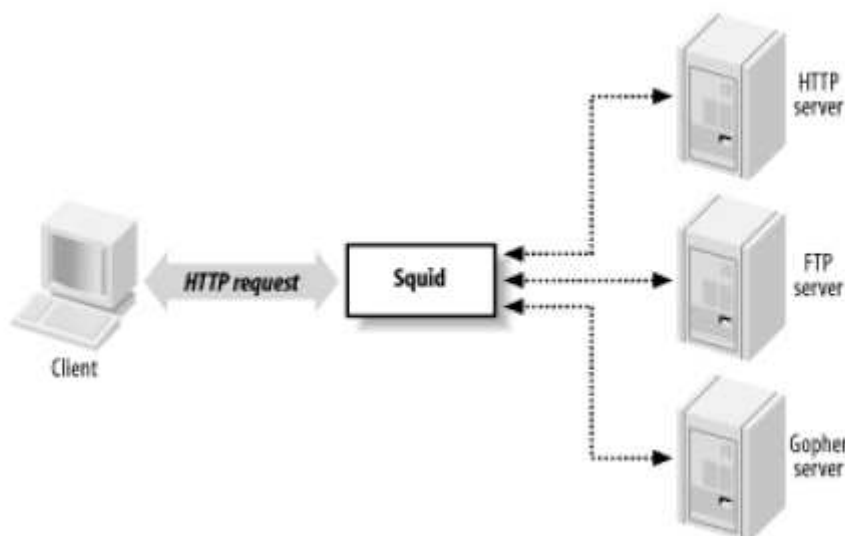


Figura 8 - Proxy Squid
Fonte: (WESSELS,2004)

No que se refere a autenticação de usuários, o Squid provê autenticação utilizando basicamente três métodos e para cada método o Squid provê módulos de autenticação também chamados de helpers:

- Método Basic – É o método onde o login trafega em claro pela rede, mas que pode utilizar vários helpers de autenticação. Como alguns exemplos:
 - o NCSA – Utiliza o usuário e senha armazenada no Linux utilizada num arquivo /etc/passwd.
 - o LDAP – Quando o usuário está em uma base de usuários de um serviço de diretórios LDAP como o OpenLDAP.
 - o SMB - é um autenticador para bases de dados Microsoft. É um programa escrito em C, que executa um shell script cada vez que se comunica com o controlador de domínio Windows. O shell script contém comandos do pacote Samba. Assim, é necessário instalar o Samba antes de utilizar o autenticador.
 - o Winbind – É um cliente para o programa winbindd do pacote Samba, para funcionar o samba deve estar instalado e configurado para um domínio além do serviço winbindd estar funcionando.
- Método NTLM – Utiliza o protocolo de autenticação NTLM já demonstrado nesse trabalho, ele utiliza principalmente dois helpers o SMB e o Winbind,

semelhantes ao método Basic. O helper Winbind é o que será utilizado na nossa implementação.

- Método Digest – Foi feito para ser mais seguro que o método Basic pois os dados de autenticação do usuário são criptografados. O helper do digest utiliza arquivos de usuários e senhas em texto claro, o que não tem suporte nativo no Linux, para utilizá-lo tem que se utilizar meios próprios para gerar esse tipo de arquivo.

2.4.2 DANSGUARDIAN

O Dansguardian é um serviço de filtro de conteúdo de páginas web. A ferramenta difere da maioria disponível no mercado pelo fato de não funcionar apenas como filtro de *URL*, mas também como um efetivo filtro de conteúdos de páginas *Web*. Pois, faz uma varredura do conteúdo de cada página acessada por seus usuários e não somente uma liberação ou proibição do nome do site ou da URL acessada, ela atua em conjunto com o Squid, atuando exatamente onde o squid não consegue prover. Atualmente ele tem suporte a autenticação NTLM e Basic, que são repassados para o squid.

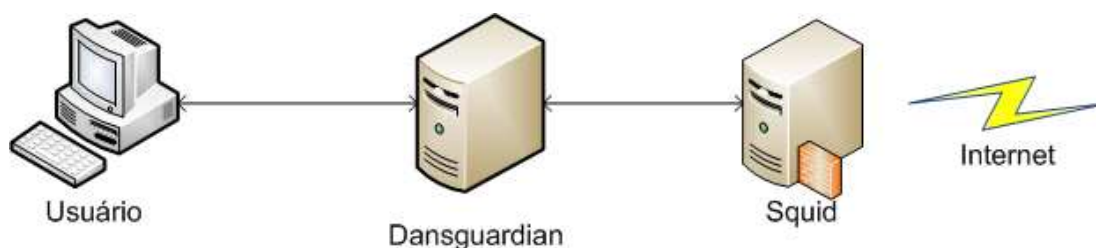


Figura 9 - Dansguardian como Intermediário entre o Proxy e o Usuário
Fonte: Próprio autor

3 DESENVOLVIMENTO DA SOLUÇÃO

Para demonstrar o desenvolvimento da solução vamos instalar dois servidores, um Windows Server 2003 R2 com o Active Directory instalado e um servidor com o Debian 5 Lenny para a instalação dos serviços necessário para o Proxy, além de um cliente com o Windows XP que ira ser um cliente do Proxy.

3.1 CONFIGURAÇÃO DO ACTIVE DIRECTORY

No Active Directory foi criado um domínio chamado CORPORACAO e dentro desse domínio criamos uma Unidade Organizacional (UO) chamada NATALVENDAS com mais três UOs: Diretoria, TI e Vendas, conforme mostrado na Figura 10.

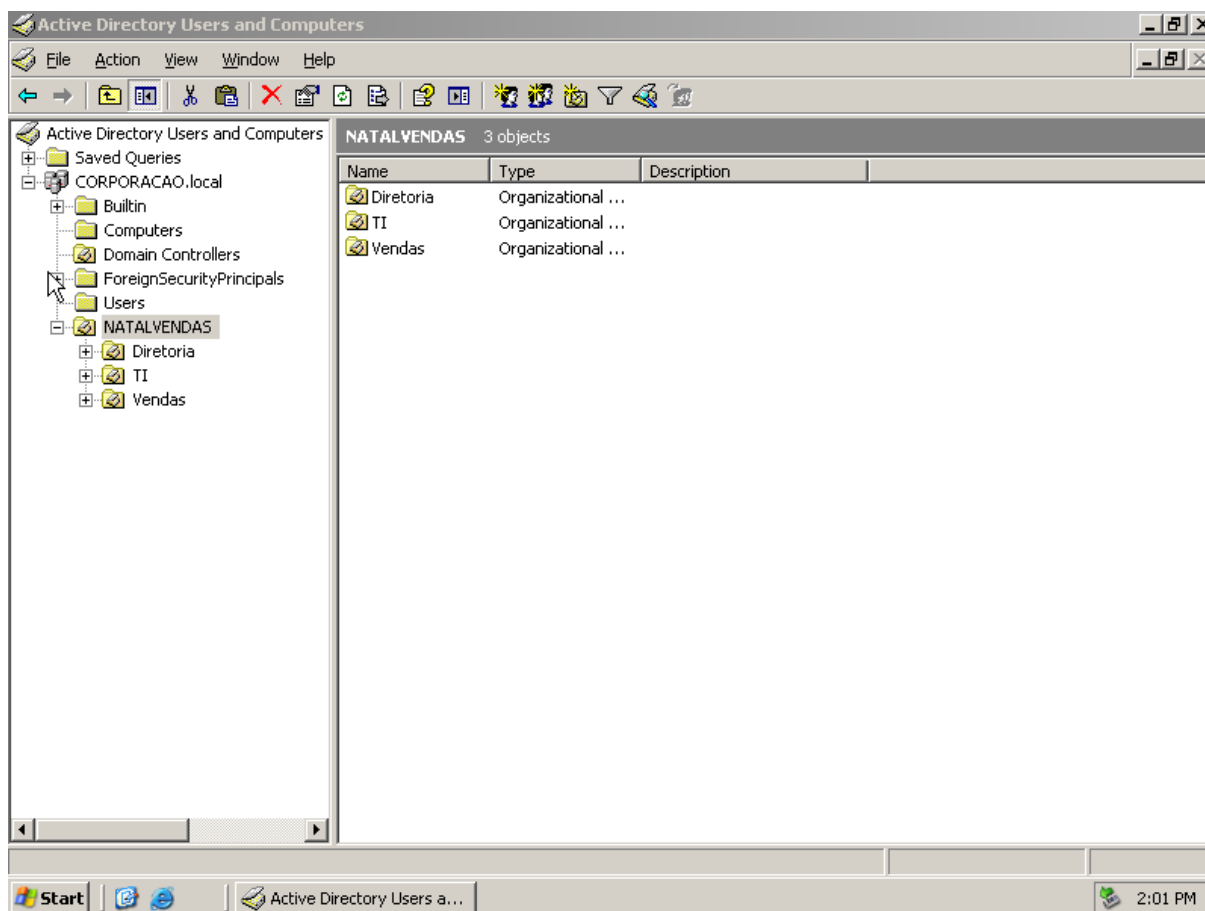


Figura 10 - Configuração do Active Directory
Fonte: Próprio autor

Foram criados três grupos de usuários, um que possui o pessoal da diretoria, um com a parte de tecnologia da informação e um com o setor de vendas, conforme mostrado na Figura 11. Esses grupos serão usados para mostrar como podemos ter configurações de Proxy para cara grupo diferente, mesmo ele usando o mesmo servidor Proxy.

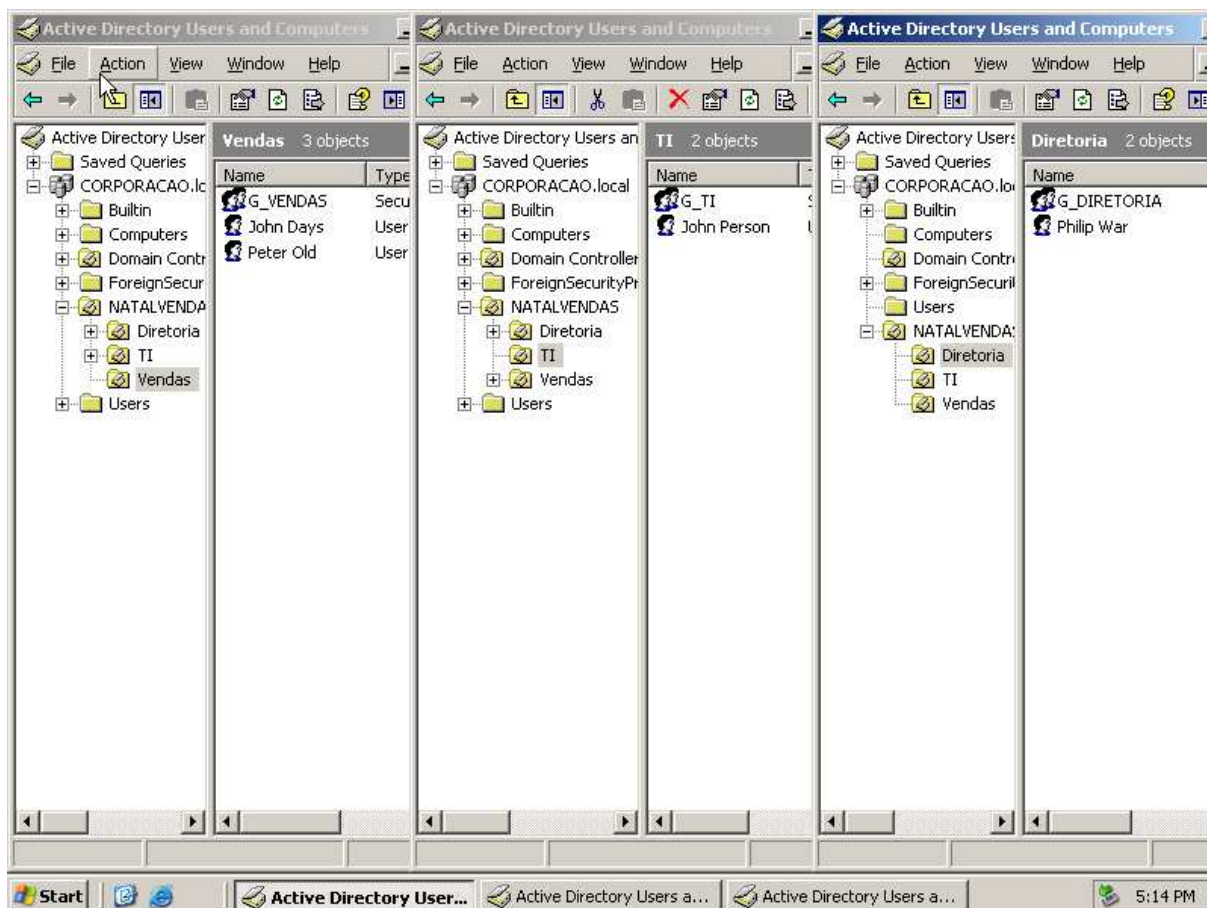


Figura 11 – Estrutura de Usuários e Grupos no AD
Fonte: Próprio autor

3.2 CONFIGURAÇÃO DO SERVIDOR PROXY

Depois de pronta a base de usuários é feita a instalação o servidor Linux que atuará como Proxy autenticado. Partindo da instalação básica de um servidor Debian e configurando uma interface de rede com um endereço que seja acessível ao AD, e é recomendado que o servidor DNS padrão do servidor Proxy seja o servidor que hospeda o AD.

Primeiro deve-se instalar os seguintes pacotes conforme a figura 12.

```
proxy:~# apt-get install winbind krb5-user ntpdate_
```

Figura 12 – Pacotes do Servidor Proxy
Fonte: Próprio autor

O pacote *winbind* é responsável pela comunicação entre o domínio Windows e o Linux utilizando NTLM, faz parte do pacote principal utilizado pelo Samba.

O pacote *krb5-user* vai fazer a comunicação entre o domínio Windows 2003 e o Linux utilizando o protocolo Kerberos.

O *ntpd* serve para sincronizar o tempo do servidor de domínio e o servidor Proxy, pois se as datas estiverem diferentes eles não conseguem se comunicar.

Quando a instalação perguntar qual o grupo de trabalho, deve ser respondido o mesmo grupo do AD, e para o servidor Kerberos o nome da máquina que está instalado o AD, conforme figura 13.

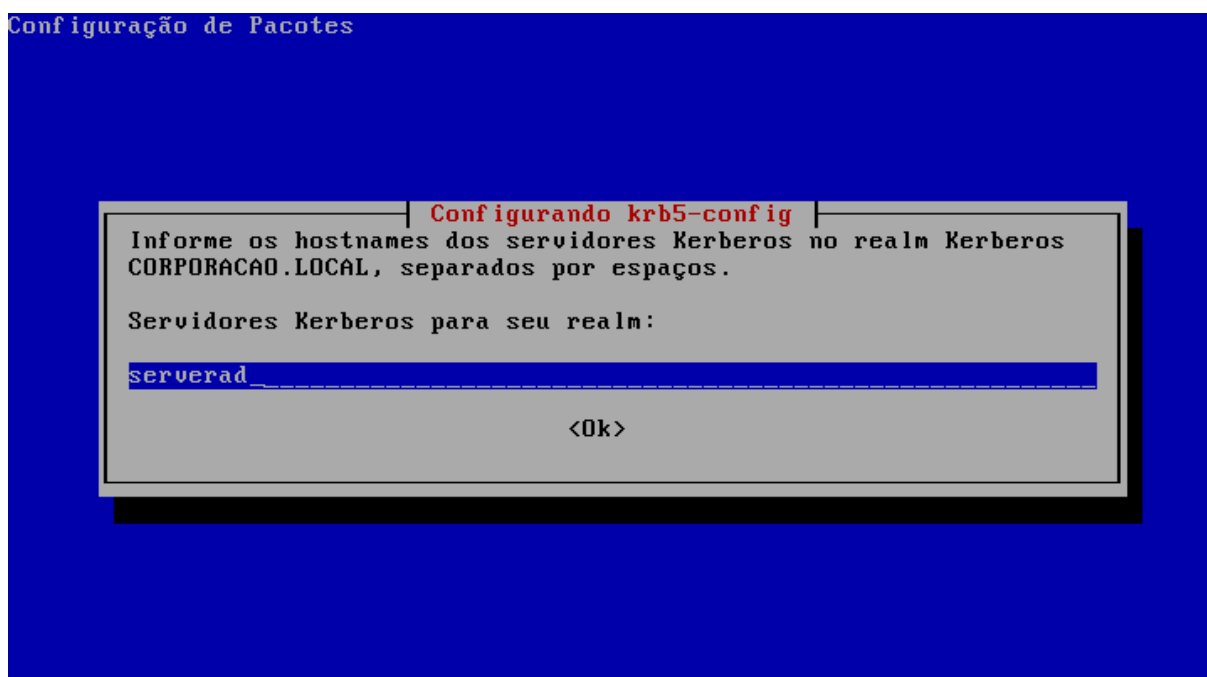


Figura 13 – Configuração para o servidor Kerberos
Fonte: Próprio autor

Deve-se agora sincronizar o relógio do servidor Proxy com o do AD usando o comando: `ntpd serverad.corporacao.local`. Em seguida alterar o arquivo `/etc/samba/smb.conf` para se conectar ao AD, conforme a Figura 14.

```

#===== Global Settings =====

[global]

## Browsing/Identification ###

# Change this to the workgroup/NT-domain name your Samba server will part of
workgroup = CORPORACAO

# server string is the equivalent of the NT Description field
server string = %h server

# Windows Internet Name Serving Support Section:
# WINS Support - Tells the NMBD component of Samba to enable its WINS Server
# wins support = no

# WINS Server - Tells the NMBD components of Samba to be a WINS Client
# Note: Samba can be either a WINS Server, or a WINS Client, but NOT both
; wins server = w.x.y.z

# This will prevent nmbd to search for NetBIOS names through DNS.
dns proxy = no

...

#### Networking ####

# interface names are normally preferred
interfaces = 192.168.1.0/24 eth1

# option cannot handle dynamic or non-broadcast interfaces correctly.
; bind interfaces only = yes
realm = CORPORACAO.LOCAL
...
##### Authentication #####

# "security = user" is always a good idea. This will require a Unix account
# in this server for every user accessing the server. See
# /usr/share/doc/samba-doc/htmldocs/Samba3-HOWTO/ServerType.html
# in the samba-doc package for details.
security = ads
password server = 192.168.1.1
# You may wish to use password encryption. See the section on
# 'encrypt passwords' in the smb.conf(5) manpage before enabling.
encrypt passwords = true
# If you are using encrypted passwords, Samba will need to know what
# password database type you are using.
passdb backend = tdbsam
obey pam restrictions = yes

...
# Some defaults for winbind (make sure you're not using the ranges
# for something else.)
idmap uid = 600-20000
idmap gid = 600-20000
; template shell = /bin/bash
winbind enum groups = yes
winbind enum users = yes
winbind use default domain = yes
# Setup usershare options to enable non-root users to share folders
# with the net usershare command.

# Maximum number of usershare. 0 (default) means that usershare is disabled.
; usershare max shares = 100

```

Figura 14- Configuração do arquivo /etc/squid/squid.conf
Fonte: Próprio autor

No arquivo `smb.conf` na seção de *Global Settings* em *workgroup* é colocado no nome do domínio Windows, no caso **CORPORACAO**.

Na seção **Network** é configurado em qual interface(ou interfaces) de rede do servidor o serviço vai estar disponível, neste exemplo foi colocada a rede 192.168.1.0/24 na interface `eth1`.

A opção *realm* é informa o domínio definido no Kerberos, no caso o domínio **CORPORACAO.LOCAL**. Essa opção tem que ser adicionada ao arquivo pois não faz parte das opções do arquivo de configuração padrão do pacote.

Na seção **Authentication** é definido os parâmetros para autenticação no AD. Primeiro coloca-se em *security = ads*, isso significa que faz parte de um domínio que utiliza o AD, no caso o Windows 2003. Coloca-se também em *password server* o IP do servidor AD. Em *encrypt passwords* deve ser informado o valor *yes* pois as senhas utilizadas são criptografadas.

Os atributos *idmap uid* e *idmap gid* são importantes, pois eles definem o intervalo de uid e gid que serão mapeados dos usuários e grupos do AD no Linux, por isso este intervalo tem que estar fora da faixa de uids e gids padrões do seu sistema para evitar conflitos de usuários. Neste caso usamos o intervalo de 600 a 10000. pois a faixa do sistema normalmente está entre 100 e 200.

Para habilitar o mapeamento entre usuários do AD e do Linux descritos no parágrafo anterior deve-se configurar os atributos *winbind enum users = yes* e *winbind enum groups = yes*.

E para os usuários não precisarem digitar o domínio toda vez que precisam se autenticar, então coloca-se o atributo *winbind use default domain = yes*.

Agora é adicionado o servidor no domínio AD utilizando o comando **net ads join -U usuário**, sendo este usuário com poderes administrativos no AD. No exemplo foi usado `administrator` (Figura 15), mas poderia ser qualquer usuário do grupo Domain Admins. Com isso, o computador será adicionado à lista de computadores do AD, conforme Figura 16.

```
proxy:~# net ads join -U administrator
Enter administrator's password:
Using short domain name -- CORPORACAO
Joined 'PROXY' to realm 'CORPORACAO.local'
proxy:~# _
```

Figura 15 - Adicionando Máquina linux no Domínio Windows
Fonte: Próprio autor

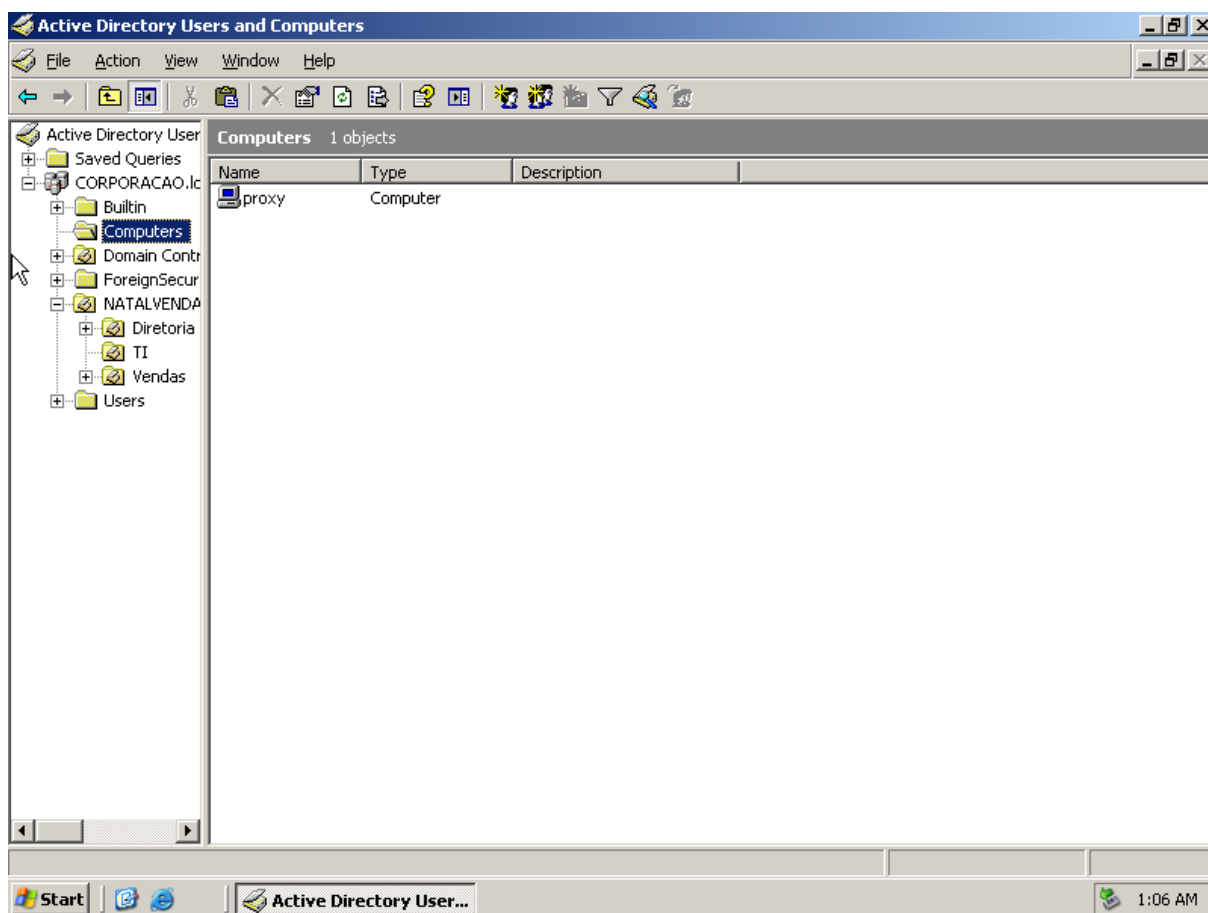


Figura 16 - Servidor Proxy cadastrado no AD
Fonte: Próprio autor

Depois disso, deve-se editar o arquivo `/etc/nsswitch.conf` para que os usuários do Windows sejam mapeados pelo Linux. Coloca-se a palavra *winbind* nas linhas que tem o *passwd* e *group* no arquivo `/etc/nsswitch.conf`, conforme Figura 17.

```
# /etc/nsswitch.conf
#
# Example configuration of GNU Name Service Switch functionality.
# If you have the 'glibc-doc-reference' and 'info' packages installed, try:
# 'info libc "Name Service Switch"' for information about this file.

passwd:          compat winbind
group:           compat winbind_
shadow:         compat

hosts:          files dns
networks:       files

protocols:      db files
services:       db files
ethers:         db files
rpc:            db files

netgroup:       nis
~
~
~
~
-- INSERT --
```

Figura 17 - Editando o arquivo /etc/nsswitch.conf
Fonte: Próprio autor

Após reiniciado o serviço Winbind é verificado se os usuários estão sendo reconhecidos usando-se o comando `wbinfo -u`, conforme figura 18.

```
proxy:~# /etc/init.d/winbind restart
Stopping the Winbind daemon: winbind.
Starting the Winbind daemon: winbind.
proxy:~# wbinfo -u
CORPORACAO\administrator
CORPORACAO\guest
CORPORACAO\support_388945a0
CORPORACAO\krbtgt
CORPORACAO\johnperson
CORPORACAO\peterold
CORPORACAO\philipwar
CORPORACAO\johndays
proxy:~# _
```

Figura 18 - Usando o `wbinfo` para listar os usuários do AD
Fonte: Próprio autor

Para comprovar se tudo está corretamente configurado, também é bom utilizar os seguintes `wbinfo -g` para listar os grupos e o `getent passwd` e `getent group`, conforme figura 19.

```
proxy:~# getent passwd | grep CORPORACAO
CORPORACAO\administrator:*:600:600:Administrator:/home/CORPORACAO/administrator:/bin/false
CORPORACAO\guest:*:601:601:Guest:/home/CORPORACAO/guest:/bin/false
CORPORACAO\support_388945a0:*:602:600:SUPPORT_388945a0:/home/CORPORACAO/support_388945a0:/bin/false
CORPORACAO\krbtgt:*:603:600:krbtgt:/home/CORPORACAO/krbtgt:/bin/false
CORPORACAO\johnperson:*:604:600:John Person:/home/CORPORACAO/johnperson:/bin/false
CORPORACAO\peterold:*:605:600:Peter Old:/home/CORPORACAO/peterold:/bin/false
CORPORACAO\philipwar:*:606:600:Philip War:/home/CORPORACAO/philipwar:/bin/false
CORPORACAO\johndays:*:607:600:John Days:/home/CORPORACAO/johndays:/bin/false
proxy:~# _
```

Figura 19 - Usando o comando `getent`
Fonte: Próprio autor

O próximo passo é instalar o squid utilizando o comando:

`apt-get install squid`

Depois da instalação edita-se o arquivo `/etc/squid.conf`. Primeiro, são descomentadas e editadas/inseridas as linhas da Figura 20 para habilitar o NTLM e utilizar o programa `wbinfo_group.pl` como um autenticador.

```
auth_param ntlm program /usr/bin/ntlm_auth --helper-protocol=squid-2.5-ntlmssp
auth_param ntlm children 15
auth_param ntlm keep_alive on
external_acl_type nt_group ttl=0 %LOGIN /usr/lib/squid/wbinfo_group.pl
```

Figura 20 - Configuração `/etc/squid.conf`
Fonte: Próprio autor

A primeira linha da Figura 20 indica que para a autenticação NTLM usamos como programa padrão `/usr/bin/ntlm_auth`.

Na segunda linha configura o número de processos filho do squid, que no exemplo é 15, e na terceira linha mantém a autenticação do usuário ativa desde a primeira autenticação.

Na quarta linha é definida um tipo de acl que usa um programa externo, no caso o `wbinfo_group.pl`, para ser chamado e fazer a verificação de permissão ou não de acesso ao proxy.

Agora pode-se inserir regras como liberação e bloqueio de usuários e grupos do AD no Squid utilizando as ACLs. Por exemplo, é possível liberar o acesso a Internet para o grupo de usuário da TI e da Direção e negar para o setor de vendas, conforme Figura 21.

```
acl Safe_ports port 280      # http-mgmt
acl Safe_ports port 488      # gss-http
acl Safe_ports port 591      # filemaker
acl Safe_ports port 777      # multiling http
acl Safe_ports port 631      # cups
acl Safe_ports port 873      # rsync
acl Safe_ports port 901      # SWAT
acl purge method PURGE
acl CONNECT method CONNECT

#inserção de acls_
acl ntlm_users proxy_auth REQUIRED
acl usuariosliberados external nt_group G_TI G_DIRETORIA
acl usuariosbloqueados external nt_group G_VENDAS

http_access deny usuariosbloqueados
http_access allow usuariosliberados
http_access allow ntlm_users

# TAG: http_access
#     Allowing or Denying access based on defined access lists
#
#     Access to the HTTP port:
-- INSERT --
```

Figura 21- Criando acls para grupos no Windows
Fonte: Próprio autor

Agora, inicia-se o serviço Squid e configura o cliente para usar o Proxy para acessar a internet, conforme Figura 22.

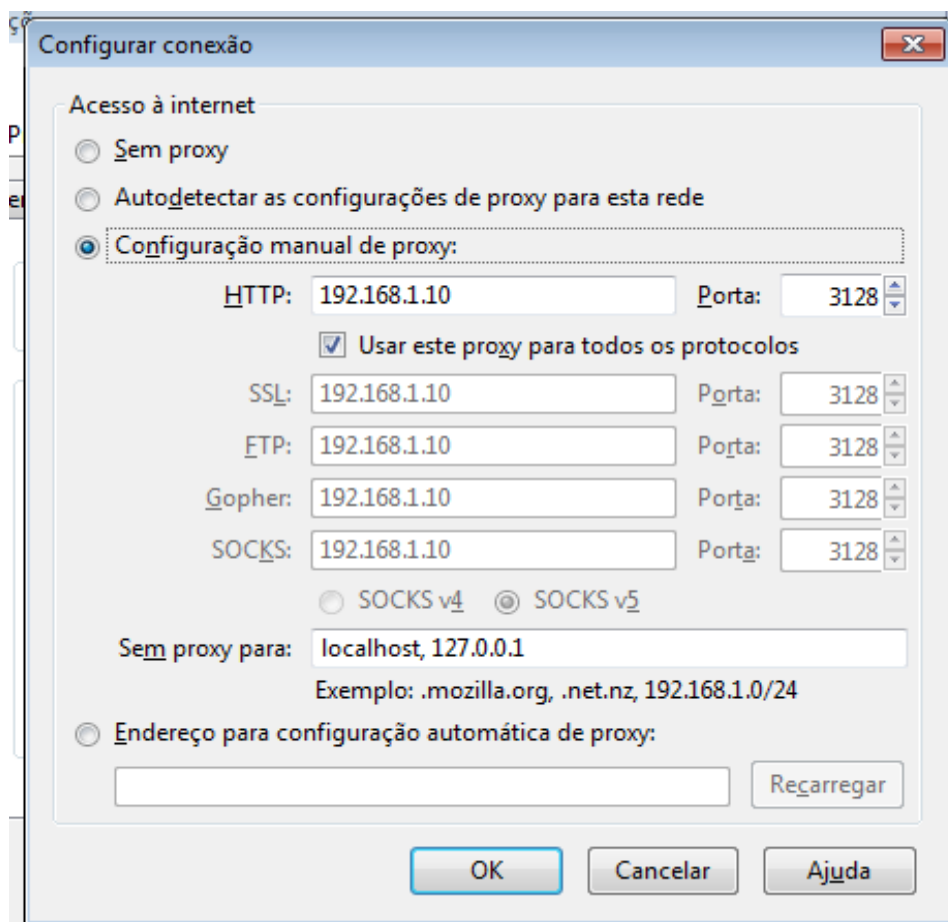


Figura 22 - Configurando o proxy no cliente
Fonte: Próprio autor

Com esta configuração o cliente já está pronto para utilizar o Proxy. Se a máquina que o cliente usa pertencer ao domínio `Corporacao.local`, a autenticação se dará de forma transparente, ou seja, sem perguntar ao usuário o login e a senha para autenticar. Se ele não pertence ao domínio, então o navegador abrirá um prompt perguntando o usuário e a senha do domínio para poder utilizar o Proxy de acordo com suas permissões.

3.3 GERANDO RELATÓRIOS DE ACESSOS USANDO O SARG

Para gerar relatórios do acesso dos usuários pode-se instalar o software livre SARG que produz relatórios como páginas web. Para instalá-lo é necessário o software Apache rodando no próprio servidor proxy. Para instalar os dois softwares, pode-se usar o comando:

```
apt-get install apache2 sarg
```

Após instalado, pode-se alterar algumas configurações dos relatórios, como idioma, editando-se o arquivo `/etc/squid/sarg.conf`, conforme Figura 23.

```
##          Russian_UFT-8
##          Russian_windows1251
##          Serbian
##          Slovak
##          Spanish
##          Turkish
##
language Portuguese

# TAG:  access_log file
#       Where is the access.log file
#       sarg -l file
#
access_log /var/log/squid/access.log

# TAG:  graphs yes|no
#       Use graphics where is possible.
#       graph_days_bytes_bar_color blue|green|yellow|orange|brown|red
#
#graphs yes
#graph_days_bytes_bar_color orange

# TAG:  title
#       Specify the title for html page.
```

Figura 23 - Configuração do sarg.conf
Fonte: Próprio autor

Nas duas linhas descomentadas da Figura 23, o SARGI é configurado para gerar o relatório em Português e o caminho do arquivo de log do squid para usar é o `/var/log/squid/access.log`

Para gerar um relatório de uso dos usuários, usa-se o comando `sarg` e em seguida a partir de um navegador, abre-se o endereço `http://ipdoservidor/squid-reports`, conforme figura 24.

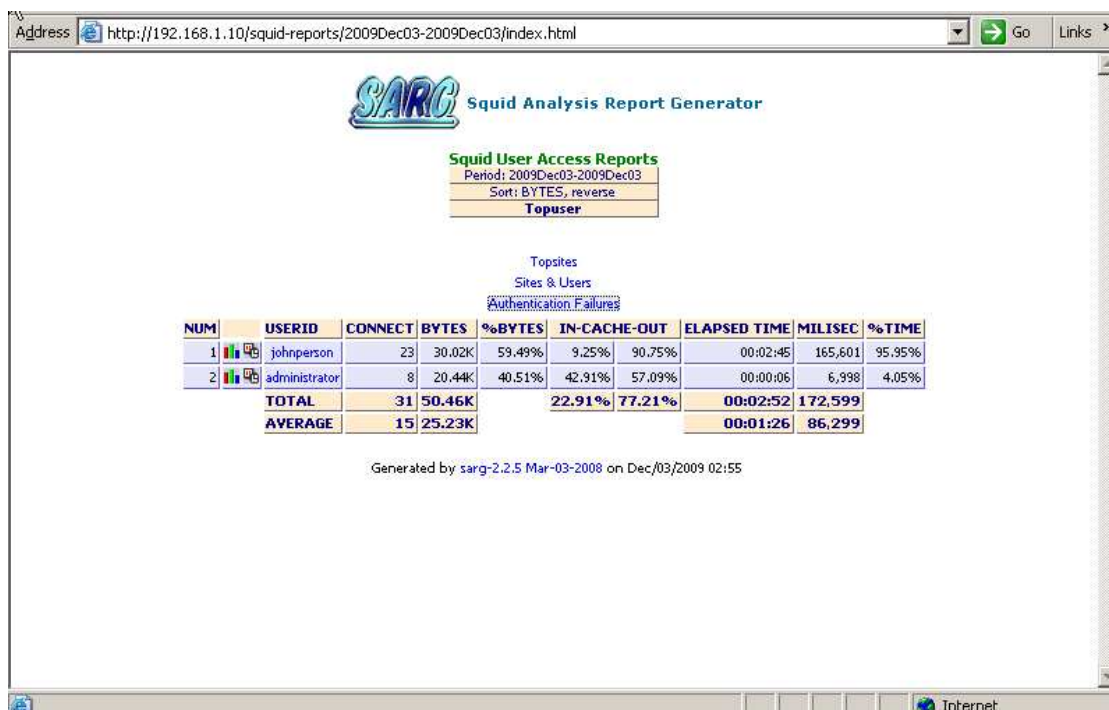


Figura 24 - Relatório usando SARG
 Fonte: Próprio autor

3.4 DANSGUARDIAN

O Dansguardian é um filtro de conteúdo que é utilizado em conjunto com o Squid, fazendo uma varredura na página web verificando seu conteúdo, e liberando ou bloqueando de acordo com o caso.

Para fazer isso o Dansguardian contém várias listas de bloqueios que podem utilizar vários tipos de bloqueios. Algumas destas listas:

- Listas negras (blacklists) – bloqueia um domínio por completo, por exemplo, google.com, ou partes destes domínios (wikipedia.com.br/vírus), ou por IP de uma máquina interna (10.222.222.222) ou externa.
- Lista de Palavras (phraselists) – bloqueio de qualquer palavra no site ou utilizando uma espécie de pesos por cada palavra encontrada.
- Expressões Regulares em URLs.
- Bloqueio por extensão: .exe, .cmd, .zip

O pacote de distribuição padrão do Dansguardian no Debian não contém a implementação do protocolo NTLM, por isso deve ser baixado o código fonte e compilado o programa para combinar com o NTLM do Squid.

Primeiro foi baixado através do software `wget` o arquivo com o código fonte mais estável e armazenado na pasta `/usr/local/src`, conforme Figura 25.

```
proxy:~# wget http://dansguardian.org/downloads/2/Stable/dansguardian-2.10.0.3.tar.gz -P /usr/local/src
--2010-01-04 06:26:32-- http://dansguardian.org/downloads/2/Stable/dansguardian-2.10.0.3.tar.gz
Resolvendo dansguardian.org... 89.16.172.190, 2001:41c8:1:5847::2
A conectar dansguardian.org[89.16.172.190]:80... conectado!
HTTP requisição enviada, aguardando resposta... 200 OK
Tamanho: 512695 (501K) [application/x-gzip]
A gravar em: '/usr/local/src/dansguardian-2.10.0.3.tar.gz'

100%[=====>] 512.695      87,9K/s   em 6,1s

2010-01-04 06:26:44 (82,3 KB/s) - '/usr/local/src/dansguardian-2.10.0.3.tar.gz'
gravado [512695/512695]

proxy:~# _
```

Figura 25 - Baixando o Pacote do Dansguardian
Fonte: Próprio autor

Para que se possa compilar o dansguardian utilizando os parâmetros que serão utilizados, devem ser instalados os pacotes `g++`, `make`, `zlib1g-dev`, `pkg-config`, `libpcre3-dev` usando o comando:

```
apt-get install g++ make zlib1g-dev pkg-config libpcre3-dev.
```

Agora, a partir da pasta onde o arquivo foi armazenado e descompactado, é usado um script do pacote chamado *configure* incluindo as opções *--enable-ntlm* (para habilitar o ntlm), *--with-proxyuser=proxy --with-proxygroup=proxy* (usuário e grupo que vai executar o daemon), *--sysconfdir=/etc* (pasta padrão com os arquivos de configurações do sistema) – *with-logdir=/var/log/dansguardian* (onde armazena os logs de acesso), *--bindir=/sbin* (pasta padrão do sistema para armazenar programas do super usuário), conforme Figura 26.

```

proxy:~# cd /usr/local/src/
proxy:/usr/local/src# tar -xzf dansguardian-2.10.0.3.tar.gz
proxy:/usr/local/src# cd dansguardian-2.10.0.3
proxy:/usr/local/src/dansguardian-2.10.0.3# ./configure --enable-ntlm --with-pro
xyuser=proxy --with-proxygroup=proxy --sysconfdir=/etc --with-logdir=/var/log/da
nsguardian --bindir=/usr/sbin_

```

Figura 26 - Configurando arquivos para compilação
Fonte: Próprio autor

Deve-se fazer a compilação do software usando o programa *make*, e se a compilação foi bem sucedida, usa-se o comando *make install* para instalar os softwares nos seus devidos diretórios.

Em seguida, é alterado o proprietário da pasta onde o arquivo de log foi instalado para o usuário que vai executar o daemon, conforme Figura 27.

```

proxy:/usr/local/src/dansguardian-2.10.0.3# chown proxy:proxy /var/log/dansguard
ian/_

```

Figura 27 - Alterando a permissão da pasta de log
Fonte: Próprio autor

Foi criado um *script shell* para ser o *daemon* do Dansguardian e o mesmo foi utilizado para iniciar o software durante a iniciação do sistema. Então, foi criado um arquivo */etc/init.d/dansguardian* (Figura 28), adicionada a permissão para execução e adicionado-o à iniciação do sistema. (Figura 29).

```
#!/bin/sh
#script startup para Dansguardian
#PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin
DAEMON=/usr/local/sbin/dansguardian
NAME=dansguardian
DESC="DansGuardian"
CONFFILELOCATION=/etc/dansguardian/
set -e
case "$1" in
start)
echo -n "Iniciando $DESC: "
sleep 3
$DAEMON
echo "$NAME."
;;
stop)
echo -n "Parando $DESC: "
$DAEMON -q
echo "$NAME."
;;
reload)
echo "Recarregando os arquivos de configuracao do $DESC."
$DAEMON
;;
restart|force-reload)
echo -n "Reiniciando $DESC: "
$DAEMON -Q
echo "$NAME."
;;
*)
N=/etc/init.d/$NAME
echo "Utilizar: $N {start|stop|restart|force-reload}" >&2
exit 1
;;
esac
exit 0
```

Figura 28 - Arquivo /etc/init.d/dansguardian
Fonte: Próprio autor

```
proxy:~# chmod +x /etc/init.d/dansguardian
proxy:~# update-rc.d -f dansguardian defaults 40
update-rc.d: warning: /etc/init.d/dansguardian missing LSB information
update-rc.d: see <http://wiki.debian.org/LSBInitScripts>
Adding system startup for /etc/init.d/dansguardian ...
/etc/rc0.d/K40dansguardian -> ../init.d/dansguardian
/etc/rc1.d/K40dansguardian -> ../init.d/dansguardian
/etc/rc6.d/K40dansguardian -> ../init.d/dansguardian
/etc/rc2.d/S40dansguardian -> ../init.d/dansguardian
/etc/rc3.d/S40dansguardian -> ../init.d/dansguardian
/etc/rc4.d/S40dansguardian -> ../init.d/dansguardian
/etc/rc5.d/S40dansguardian -> ../init.d/dansguardian
proxy:~# _
```

Figura 29 - Adicionando o Daemon na inicialização do sistema
Fonte: Próprio autor

Para as configurações do arquivo /etc/dansguardian.conf, na Figura 30, em *filterip* é onde se coloca em qual endereço IP das interfaces de rede que o *daemon* vai escutar. Se desejar escutar em todas as interfaces, deixa-se o mesmo em branco. O outro parâmetro

filterport é a porta TCP que está será usada. Em *proxyip* e *proxyport* são informados o endereço e a porta que estão sendo usados pelo Squid.

```
# Network Settings
#
# the IP that DansGuardian listens on. If left blank DansGuardian will
# listen on all IPs. That would include all NICs, loopback, modem, etc.
# Normally you would have your firewall protecting this, but if you want
# you can limit it to a certain IP. To bind to multiple interfaces,
# specify each IP on an individual filterip line.
filterip =

# the port that DansGuardian listens to.
filterport = 8080

# the ip of the proxy (default is the loopback - i.e. this server)
proxyip = 127.0.0.1

# the port DansGuardian connects to proxy on
proxyport = 3128

# accessdeniedaddress is the address of your web server to which the cgi
# dansguardian reporting script was copied. Only used in reporting levels 1 and
# 2.
#
# This webserver must be either:
```

Figura 30 - Configuração do arquivo /etc/dansguardian.conf - Parte 1
Fonte: Próprio autor

Em seguida, é habilitado o plugin de autenticação do NTLM , conforme Figura 31.

```
# Auth plugins
# These replace the usernameidmethod* options in previous versions. They
# handle the extraction of client usernames from various sources, such as
# Proxy-Authorisation headers and ident servers, enabling requests to be
# handled according to the settings of the user's filter group.
# Multiple plugins can be specified, and will be queried in order until one
# of them either finds a username or throws an error. For example, if Squid
# is configured with both NTLM and Basic auth enabled, and both the 'proxy-basic
# and 'proxy-ntlm' auth plugins are enabled here, then clients which do not supp
ort
# NTLM can fall back to Basic without sacrificing access rights.
#
# If you do not use multiple filter groups, you need not specify this option.
#
#authplugin = '/etc/dansguardian/authplugins/proxy-basic.conf'
#authplugin = '/etc/dansguardian/authplugins/proxy-digest.conf'
#authplugin = '/etc/dansguardian/authplugins/proxy-ntlm.conf'
#authplugin = '/etc/dansguardian/authplugins/ident.conf'
#authplugin = '/etc/dansguardian/authplugins/ip.conf'

-- INSERT --
```

Figura 31 - Configuração do arquivo /etc/dansguardian.conf - Parte 2
Fonte: Próprio autor

Os clientes do dansguardian o utilizam pelo navegador de forma semelhante ao utilizado pelo squid, conforme configuração no cliente vista na Figura 22. Somente é alterada a porta e o endereço do servidor onde está instalado o Squid, como visto na Figura 32.

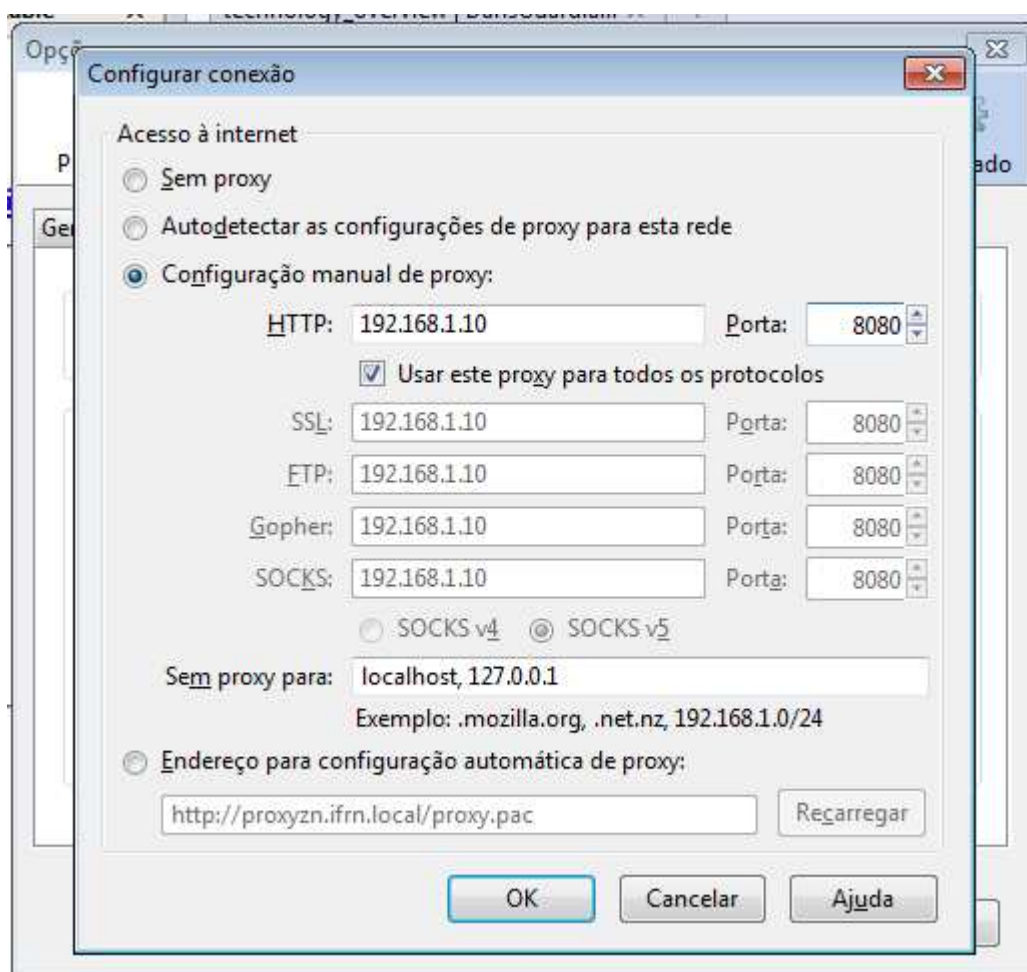


Figura 32 - Alteração do navegador pelo usuário
Fonte: Próprio autor

4 CONCLUSÃO

Este trabalho mostrou como utilizar usuários do Windows em Serviços Linux fazendo com que podemos optar em utilizar a interoperabilidade entre sistemas e utilizá-las de acordo com as nossas necessidades. Utilizando as vantagens de cada um dos sistemas como a centralização de informações do AD com a estabilidade e segurança do Linux. Além dos exemplos utilizados neste trabalho podemos ver a interoperabilidade em sistemas de e-mail, servidores RADIUS, autenticação em páginas PHP e qualquer outra aplicação que possa utilizar uma base LDAP.

Para que a autenticação seja obrigatória deve-se criar regras no firewall para que a conexão para a internet na porta HTTP e HTTPS somente seja feita pelo servidor Proxy.

REFERÊNCIAS

CALÔR FILHO, M. D. **Kerberos**. Disponível em <http://www.gta.ufrj.br/grad/99_2/marcos/kerberos.htm>. Acessado em 29 ago. 2009.

CORPORATION, M. **Windows 2003**. 2003. Disponível em <http://download.microsoft.com/download/E/F/4/EF47DD19-6E9D-426F-A871-E0730C8B2EA9/WindowsServer2003_LC.zip>. Acessado em 25 ago. 2009.

SILVA, Hedio Carvalho E. **Solução de controle de acesso a Internet em ambientes corporativos**. 2005. Monografia (Especialização em Administração de Redes Linux) – Universidade Federal de Lavras, Lavras, 2005.

TSCHALÄR, Ronald. **NTLM**, Disponível em <<http://www.innovation.ch/personal/ronald/ntlm.html>>. Acessado em 20 ago. 2009.

WESSELS, D. **Squid: the definitive guide**. [S.l.]: O'Reilly Media, 2004.

Jesus, Alessandro Erik de

Implantação de autenticação de serviços Linux no
serviço de diretório active directoty / Alessandro Erik de
Jesus. ____ Natal: FARN, 2010.

42 p.

Orientador: Aluizio Ferreira da Rocha Neto, M.Sc.

Monografia (Especialização em Redes de
Computadores)-Faculdade Natalense para o
Desenvolvimento do Rio Grande do Norte.

1. Redes de computadores – monografia 2.
Autenticação de usuários - informática – monografia. I.
Título

RN/FARN/BC

CDU 004.7

**LIGA DE ENSINO DO RIO GRANDE DO NORTE
FACULDADE NATALENSE PARA O DESENVOLVIMENTO DO RN
ESPECIALIZAÇÃO EM REDES DE COMPUTADORES**

**IMPLANTAÇÃO DE AUTENTICAÇÃO DE SERVIÇOS LINUX UTILIZANDO
SERVIÇO DE DIRETÓRIO ACTIVE DIRECTORY**

ALESSANDRO ERIK DE JESUS

**NATAL/RN
2010**