ALEXSANDRO QUEIROZ DA SILVA

Implantação de servidor *proxy* utilizando o Squid em modo não autenticado, junto com o DansGuardiam para controle de conteúdo na instituição de ensino

Monografia de Pós-Graduação "*Lato Sensu*" apresentada ao Departamento de Ciência da Computação para obtenção do título de Especialista em "Administração em Redes Linux"

Orientador
Prof. Joaquim Quinteiro Uchoa

LAVRAS
MINAS GERAIS - BRASIL
2010

ALEXSANDRO QUEIROZ DA SILVA

Implantação de servidor *proxy* utilizando o Squid em modo não autenticado, junto com o DansGuardiam para controle de conteúdo na instituição de ensino

Monografia de Pós-Graduação "*Lato Sensu*" apresentada ao Departamento de Ciência da Computação para obtenção do título de Especialista em "Administração em Redes Linux"

Aprovada em *04 de Dezembro de 2010*

Prof. Herlon Ayres Camargo

Prof. Tales Heimfarth

Prof. Joaquim Quinteiro Uchoa
(Orientador)

LAVRAS
MINAS GERAIS - BRASIL
2010

Dedico a meus irmãos Patrícia, Daniela, Fábio, minha amada Vilany, a bela Isabella filha que tanto amo, meus pais Iolanda e Eustábio, que mostrou a importância dos estudos em minha vida. E principalmente a Deus que tornou isso possível. E a todos aqueles que me deram força nos momentos em que eu mais precisava.

Agradecimentos

Agradeço a Deus por nos guiar sempre em todos momentos de nossas vidas. Ao orientador, Prof. Joaquim Quinteiro Uchoa, pela dedicação e comprometimento dispensados durante a realização do trabalho. A todos os demais Professores pelas instruções e direcionamento do curso .

Sumário

Lista de Figuras

Lista de Tabelas

Resumo

Diante da necessidade do controle de acesso a Internet para otimizar a utilização da banda desse meio de comunicação com mais segurança do conteúdo visitado, na qual esta ligada diretamente com a qualidade dos serviços prestados, o presente estudo propõe mostrar a implantação de servidor *proxy* com o Squid como solução de performance, junto com o DansGuardiam para controle de conteúdo na instituição Centro Educacional Casa do Estudante no município de Itabatã-BA .

Palavras-Chave: Servidor *Proxy*; Squid; DansGuardian; Controle de Conteúdo.

Capítulo 1

INTRODUÇÃO

1.1 Justificativa

A tecnologia cresce vertiginosamente sendo um recurso extremamente importante para se conseguir aliar qualidade, agilidade e melhoria dos serviços prestados. A Internet que já se solidificou como instrumento de comunicação, ensino e pesquisa, há vários anos tornou-se a base sobre a qual se desenvolvem e se agregam tecnologias novas ou previamente existentes.

A agilidade e facilidade nas trocas de informações desse meio tornam-se cada vez mais eficientes e aumentam a competitividade das corporações, influenciando positivamente inclusive diversos setores, como, por exemplo, do ensino e até mesmo o governo. É cada vez mais comum o ensino realizado através de meios eletrônicos, e o comercio eletrônico é uma realidade presente no cotidiano de diversas instituições, configurando o que hoje se denomina a "Sociedade da Informação". (NAKAMURA; GEUS, 2007).

No entanto, paralelamente as vantagens provenientes do uso cada vez mais difundido da Internet, os crimes que são cometidos através deste meio de comunicação aberta também crescem enormemente, emergindo assim uma crescente preocupação com a segurança de informações. O acesso por pessoas não autorizadas a informações sigilosas podem causar enormes prejuízos a uma empresa ou usuário, o que tem direcionado diversos esforços no sentido de melhorar a segurança dos sistemas computacionais envolvidos, visando impedir, ou pelo menos limitar, o acesso não autorizado.

Esses fatos evidenciam a grande necessidade para se investir mais com a segurança das redes.

Os gráficos das figuras 1.1, 1.2, 1.3, mostram que o número de incidentes tem aumentado espantosamente nos últimos anos, chegando a quase dobrar de 2008 para 2009 como são evidenciados na Figura 1.2. Para maior entendimento da Figura 1.1, segue-se um detalhamento de sua legenda:

- ***Worm***: São programas maliciosos que se propagam num processo automatizado na rede os códigos maliciosos.
- ***DoS*** (DoS – Denial of Service): São ataques de negação de serviço que não são criados com o propósito de invasão de sistemas, mas com o objetivo de torná-lo indisponível .
- **Invasão**: Um ataque bem sucedido que resulta no acesso não autorizado a um computador ou rede.
- ***Web***: Ataque que visa o comprometimento de servidores *Web* ou desfigurações de páginas na Internet.
- ***Scan***: Notificações de varreduras em redes de computadores com o intuito de identificar quais computadores estão ativos e quais serviços estão sendo disponibilizados por eles.
- **Fraude**: É o ato de enganar os outros com propósito de predicá-los ou até mesmo obter propriedade ou serviços.
- **Outros**: Notificações de incidentes que não se enquadram nas categorias anteriores.

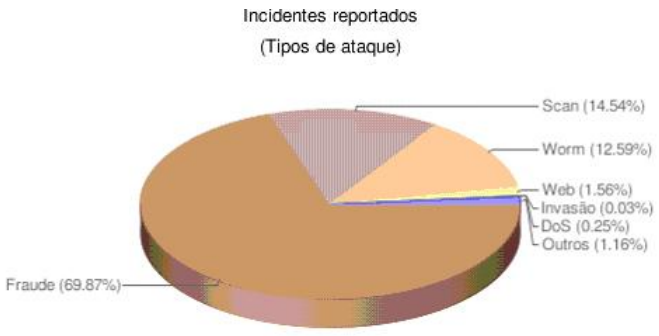


Figura 1.1: Incidentes Reportados ao CERT.br de Janeiro a Dezembro 2009. Fonte: (CERT.BR, c).

CERT.BR2010 .

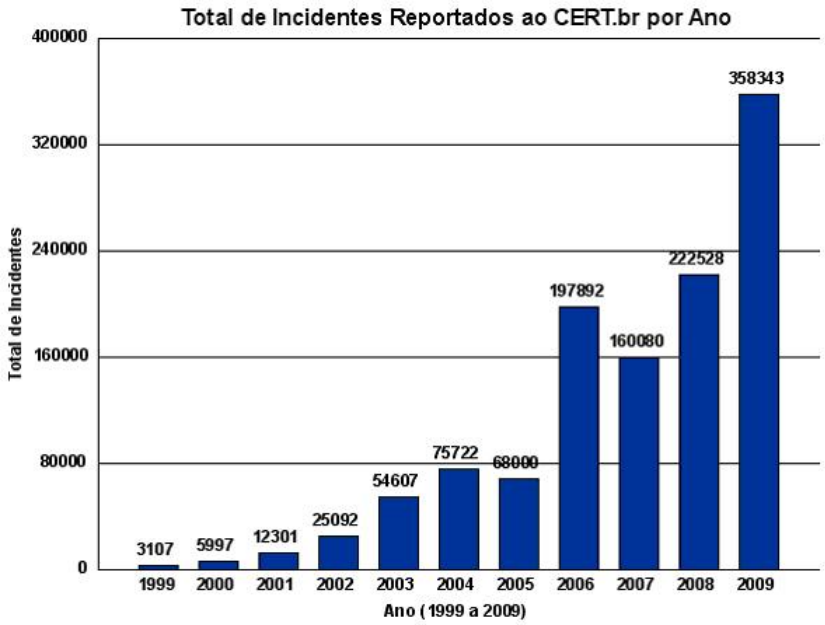


Figura 1.2: Estatísticas dos Incidentes Reportados ao CERT.br por ano. Fonte: (CERT.BR, a)

Diante desse contexto, existia uma grande preocupação do Centro Educacional Casa do Estudante Colégios Leon Feffer I e II, localizado no município de Itabatã, estado da Bahia, com a fragilidade da segurança da informação em sua rede de computadores visando minimizar os riscos de vírus, *trojans*, "vazamento" de informações confidenciais e outras pragas da Internet. Esses fatos evidenciaram a importância do controle eficiente na rede da Instituição na busca da melhoria no desempenho desse meio de comunicação, aumentando assim a produtividade e evitar os problemas legais causados pelo uso impróprio do computador, como casos de *downloads* de conteúdos ilegais, provenientes, por exemplo, de pirataria ou ainda pornografia infantil.

Conforme o Art. 932, III, do Código Civil Brasileiro (BRASIL, 2002), o empregador, é responsável pela prática de atos cometidos pelos seus empregados, dentro do ambiente corporativo " o patrão, amo ou comitente, por seus empregados, serviçais e prepostos, no exercício do trabalho que lhes competir, ou por ocasião dele".

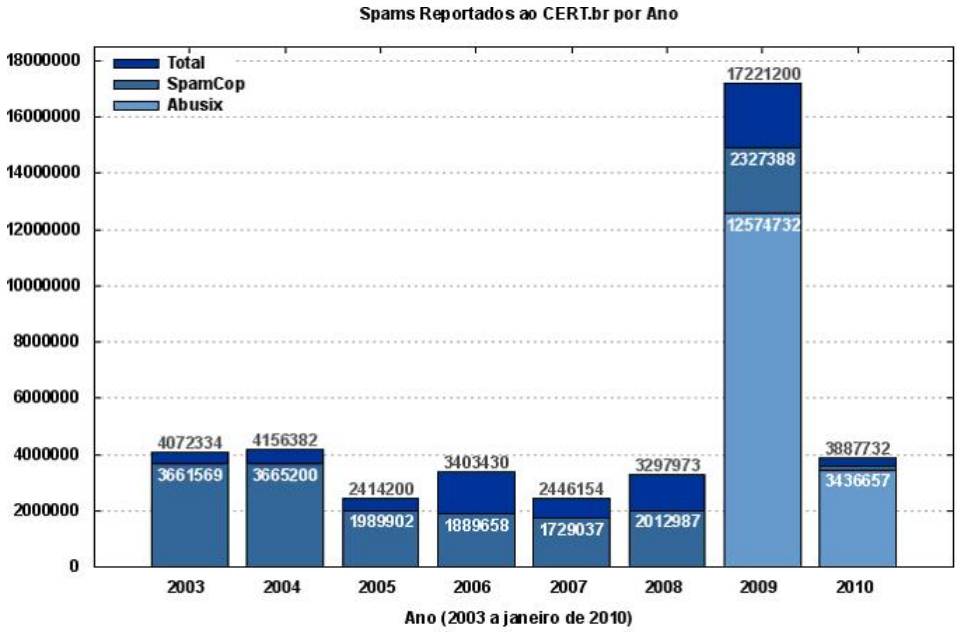


Figura 1.3: Estatísticas de Notificações de Spam Reportadas ao CERT.br. Fonte: (CERT.BR, b)

Sendo assim, tratar prontamente, racionalmente e com segurança as informações era um dos grandes desafios dessa instituição.

Devido a transtornos em que a falta de controle do conteúdo na rede, estava ocasionando para a empresa, surgiu à motivação para implantação de servidor *proxy* utilizando o Squid, junto com o DansGuardiam para controle de conteúdo na instituição Centro Educacional Casa do Estudante Ltda.

1.2 Objetivos

1.2.1 Objetivo Geral

O presente trabalho tem como objetivo geral a adoção de *software* livre para otimizar a utilização da banda de Internet com mais segurança do conteúdo visitado, na qual está ligada diretamente com a qualidade dos serviços prestados na

Instituição de ensino Centro Educacional Casa do Estudante.

1.2.2 Objetivos Específicos

- implantação de servidor *proxy* com o Squid não autenticado, para reduzir a utilização da conexão e melhorar os tempos de resposta fazendo *cache* de requisições frequentes de páginas *Web*, como solução de performance;
- Bloqueio do conteúdo impróprio contido nas páginas da Internet com o uso do DansGuardian na rede de computadores na instituição em estudo, visando propiciar um maior controle das informações através de um tratamento com mais segurança, agilidade e mais qualidade dos serviços.

1.3 Metodologia

Para atingir os objetivos desse trabalho foi realizada uma pesquisa bibliográfica com base em teorias e métodos científicos de autores relacionados ao tema em fundamento.

Contextualizada a parte teórica, a parte prática foi realizada por meio da análise da estrutura física e humana do ambiente onde o trabalho está inserido, bem como, através da instalação das ferramentas necessárias, configurações e testes para adequação ao ambiente da Instituição Centro Educacional Casa do Estudante Ltda, que serão detalhadas nos capítulos seguintes.

1.4 Estrutura

O trabalho encontra-se estruturado em cinco capítulos: o segundo capítulo aborda sobre o histórico, a contextualização e a problematização, bem como, traz uma apresentação geral sobre a empresa abordada no estudo, o problema e solução adotada. O terceiro capítulo descreve sobre as ferramentas utilizadas direta e/ou indiretamente no desenvolvimento do presente trabalho, ressaltando também a importância e o objetivo das mesmas. O quarto capítulo descreve as instalações e configurações das ferramentas necessárias para a implantação do presente trabalho. O quinto capítulo relata os problemas ocorridos na implantação e os resultados obtidos.

Capítulo 2

HISTÓRICO, CONTEXTUALIZAÇÃO e PROBLEMATIZAÇÃO

Este capítulo tem como finalidade fazer uma apresentação geral sobre a empresa abordada no trabalho, o problema e a possível solução.

2.1 Histórico

O Centro Educacional Casa do Estudante LTDA é uma empresa de capital privado com sede no município de Aracruz no Estado do Espírito Santo. Mantém um contrato em regime de comodato com a Suzano Papel e Celulose S.A. para realizar os serviços de operação/administração das unidades escolares Leon Feffer I e II , fundadas respectivamente em 1991 e 1992, para atender, prioritariamente, às necessidades escolares dos filhos dos colaboradores da antiga Bahia Sul Celulose S.A., atualmente Suzano Papel e Celulose S.A., sua mantenedora, no município de Mucuri, extremo sul da Bahia.

Em 1998 os serviços de operacionalização e de funcionamento desses colégios objetos de licitação realizada pela mantenedora, foram adjudicados ao Centro Educacional Casa do Estudante Ltda. Com competência para contratar e dispensar todo o quadro de pessoal, assim como, definir metodologia a ser utilizada pelos docentes, escolher e adotar o material didático que estiver de acordo com a filosofia de trabalho da instituição.

Importante ressaltar que o Centro Educacional Casa do Estudante representa um total de quatro unidades escolares, a saber: a sede em Aracruz e uma filial no município de João Neiva (Espírito Santo), os Colégios Leon Feffer I e II, respectivamente no distrito de Itabatã e na sede do município de Mucuri (Bahia). Para os fins deste trabalho, será considerada apenas a unidade Itabatã. A referência feita a estes será por meio da utilização dos vocábulos colégios ou unidades escolares e instituição ou escola quando a referência for ao Centro Educacional Casa do Estudante — unidade administrativa.

Os colégios oferecem cursos de Educação Infantil, Ensino Fundamental e Médio com pré-vestibular integrado, funcionando nos turnos matutino e vespertino, em regime de externato para ambos os gêneros, e têm como objetivo geral garantir ao estudante a formação indispensável dos conhecimentos necessários para desenvolver e exercitar a cidadania e a continuidade dos estudos. O Centro Educacional Casa do Estudante conta com cerca de 88 funcionários distribuidos conforme Tabela 2.2 e 1010 alunos matriculados, distribuídos quase que igualitariamente nas duas unidades escolares do município de Mucuri. Os alunos em sua grande maioria, residem no município, mas, há alguns que vêm de distritos e outros municípios vizinhos. Estes, por sua vez, estão distribuídos conforme Tabela 2.1.

Tabela 2.1: Quantitativo dos alunos do Centro Educacional Casa do Estudante Ltda

Segmentos	**Número de alunos**
Educação *Infantil*	203
Ensino Fundamental	555
Ensino médio	252

Tabela 2.2: Quantitativo de funcionários por departamento.

Departamento	**Número de funcionários**
Administrativo	23
Pedagógico	46
Limpeza e Manutenção	19

O Centro Educacional Casa do Estudante Ltda., é uma instituição de Ensino Fundamental e Médio certificada no Sistema da Gestão da Qualidade ISO 9001:2008, no qual esse projeto encontra-se inserido. Sua estrutura inicial funcionava da seguinte forma: 32 máquinas no laboratório de Informática, funcionavam com duas opções de inicializações, tais como, o Microsoft Windows XP, e o Linux

distribuição Slackware na versão 11.0 para as pesquisas e aprendizado dos Alunos, e a equipe da Tecnologia da Informação da Instituição, 14 máquinas *desktops* e *notebooks* na administração, 3 máquinas na sala dos professores e um servidor Windows 2003 Server com os seguintes serviços:

1. Servidor de Banco de Dados: hospedado o Microsoft SQL Server 2003, que é o banco de dados do Sistema de Gerenciamento Escolar, o Collegium
2. Servidor de Internet: que nesse período recebia um *link* de acesso à Internet via rádio de 1024 kbps, para compartilhar para rede.
3. Controlador de Domínio
4. Servidor de impressão

2.2 Os Problemas Enfrentados

Nesse antigo cenário não havia controle do conteúdo visitado à Internet. O laboratório de informática era o departamento mais crítico, haja vista que, por ter acesso livre à Internet, as aulas estavam sendo prejudicadas devido aos desvios de atenções dos alunos com conteúdos impróprios, *downloads* de arquivos, bem como, programas proibidos e nocivos, que danificavam as máquinas da Instituição e conseqüentemente aumentavam os custos de manutenção das mesmas. Devido a grande quantidade de serviços instalado em uma única máquina, o servidor travava constantemente. Nas reuniões de pais, a preocupação com o conteúdo visitado pelos seus filhos no laboratório de Informática era palco de grandes discussões. De acordo com a Lei No 8.069, de 3 de julho de 1990 (BRASIL, 1990), submeter criança ou adolescente a prostituição ou a exploração sexual é crime com pena de 4 a 10 anos e multa.

Outro grande problema enfrentado pela instituição era o alto consumo de banda com arquivos e programas como filmes, músicas, e vídeos clipes por parte dos funcionários e alunos. Em consequencia, o desempenho nos acessos à Internet era drasticamente reduzido. Dessa forma, a instituição sentiu a necessidade de melhorar a eficiência desses serviços. O que desencadeou uma pesquisa para identificar as ferramentas desenvolvidas como software livre para melhorar o controle de acesso a Internet e otimizar a utilização da banda com mais segurança.

Capítulo 3

FERRAMENTAS UTILIZADAS

Neste capítulo são apresentadas as principais ferramentas que foram utilizadas direta e/ou indiretamente no desenvolvimento desse projeto para a escola Centro Educacional Casa do Estudante.

3.1 Sistema Operacional

Para implantação do presente trabalho, foi escolhido o sistema operacional Linux distribuição Slackware pelo fato, de ter apresentado flexibilidade, solidez e estabilidade nos estudos feitos no laboratório de Informática, pelo departamento de Tecnologia da Informação da Instituição. O Slackware por ser uma distribuição bastante usada no mundo, possui boa documentação disponível na Internet pela comunidade de Software Livre em listas de discussões. Segundo (RICCI, 2004), o Slackware é um sistema operacional Linux avançado. Foi projetado tendo como objetivos principais, a estabilidade e a fácil utilização e, acima de tudo, a confiabilidade do sistema operacional. Por isso, são incorporados apenas os pacotes mais estáveis e maduros disponíveis na época de sua liberação oficial.

3.2 *F*irewall Iptables

O Iptables é um *Firewall* nativo do *kernel* 2.4 e 2.6 que permite a implementação de filtros e tratamento de pacotes através de simples tabelas que funcionam baseados no endereço/porta de origem/destino do pacote, prioridade, etc. Segundo (RIBEIRO, 2004) o iptables utiliza a infraestrutura do *kernel*l através do Netfilter para saber como filtrar e até alterar dados empacotados, com base em diversos critérios.

Segue abaixo algumas características do Iptables:

- é muito rápido, seguro e bastante estável;
- suporta redirecionamento de portas que foi usado no presente trabalho para o redirecionar as requisições recebidas na porta 80 para o Squid (*proxy* em modo transparente);
- suporta *SNAT* (modificação do endereço de origem das máquinas para um único endereço ou faixa de *IPs*;
- suporta *DNAT* (modificação do endereço de destino das máquinas para um único endereço ou faixa de *IPs*) ;
- suporta interfaces de origem/destino de pacotes;
- possui mecanismos internos para rejeição automática de pacotes duvidosos ou mal formados.

Segundo (LIMA, 2000), um bom *Firewall* não deve permitir que pacotes inválidos entrem na rede a ser protegida. Os pacotes inválidos são aqueles que não satisfazem as regras de filtragem e/ou não pertencem a nenhuma conexão ou sessão existente através do firewall.

O Iptables pode ser usado para bloquear pacotes com destino a determinados endereços:
`#iptables -A FORWARD -s 192.168.200.0/24 -d www.sexo.com.br -j DROP.`

A Figura 3.1 ilustra a filtragem de conteúdo pelo *firewall*. Nesse exemplo o *firewall* nega pacotes de origem a rede LAN com destino ao *site* `www.sexo.com.br`.

Figura 3.1: *Firewall.*

Onde:

- A: Adiciona uma nova regra ao sistema.

- FORWARD: verifica pacotes que entram e saem do *firewall* na rede interna.
- -s: dados e endereço de origem.
- 192.168.200.0/24: rede interna (Origem).
- -d : Dados e endereço de destino.
- www.sexo.com.br: endereço que será bloqueado.
- -j: especifica o destino de uma regra e redireciona para ação a ser tomada.
- DROP: não permite a passagem do pacote.

No entanto, o filtro de conteúdo com o Iptables pode ser uma tarefa bastante tediosa para o Administrador de rede, considerando que as suas regras utilizada para tais fins podem aumentar bastante devido ao grande número de *sites* a serem bloqueados, e milhares que surgem todos os dias. Quando sua base de bloqueios aumenta, a administração de tais regras podem ser tornar confusas, o que leva muitas vezes a erros e a criação de brechas para ações maliciosas.

3.3 *Proxy* Squid

O Squid é um software livre o que implica dizer que ele está licenciado nos termos da *GPL (General Public License)*. Seu objetivo primário é agilizar o acesso à um conteúdo *Web* qualquer através do armazenamento em *cache* local. Um servidor *proxy* funciona como um intermediário no contato dos computadores da rede local com outras máquinas fora dela, como, por exemplo, na Internet. Ele recebe as requisições de qualquer navegador de rede por conteúdo que está no servidor de rede; esta requisição é mantida pelo Squid. Possuindo o conteúdo requisitado em seu repositório, ele verifica se está atualizado com o da Internet. O que mantém o *cache* atualizado, e evita que o cliente receba arquivos obsoletos. Se o conteúdo não está no repositório, o Squid pega o conteúdo do servidor de rede e então o serve para o cliente.

Um dos usos mais comuns de computadores com Linux em redes corporativas é o papel de elo de ligação entre os computadores locais e a Internet. Há muitas maneiras de configurar este serviço, e uma das mais populares é o uso de servidores *proxy*, que permitem o acesso aos serviços da Internet com segurança e economia de recursos (PINHEIRO, 2010).

Entre as vantagens do Servidor *Proxy* Squid, destacam-se:

1. É possível impor restrições de acesso com base no horário, *login*, endereço *IP* da máquina. Essa função dar flexibilidade ao administrador de rede que precisa deixar o conteúdo livre em determinados horários, como, nos de almoço.

2. O *proxy* funciona como um *cache* de páginas e arquivos armazenando informações já acessadas conforme ilustra a Figura 3.2, quando alguém acessa uma página que já foi carregada, o *proxy* envia os dados que guardou no *cache* sem precisar acessar a mesma página repetidamente. Isso pode gerar uma economia de banda tornando o acesso mais rápido sem precisar investir em uma conexão mais rápida.

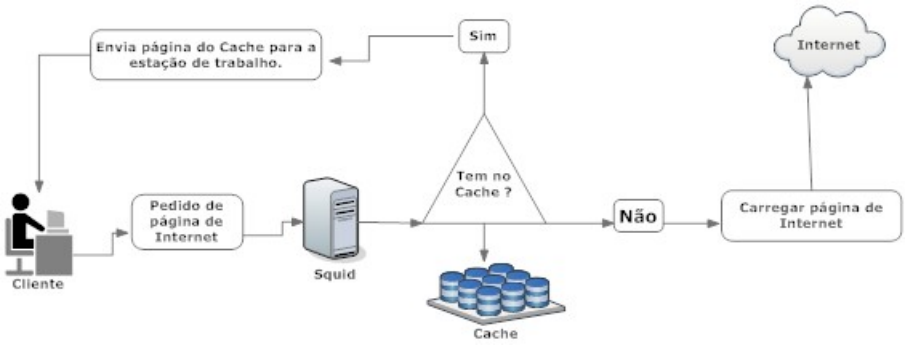


Figura 3.2: *Proxy Cache.*

3. Hoje em dia os *sites* costumam usar páginas dinâmicas onde o conteúdo muda a cada visita, mas, mesmo nestes casos, o *proxy* dá uma ajuda, pois embora o conteúdo HTML da página seja diferente a cada visita e realmente precisa ser baixado de novo, muitos componentes da página, como ilustrações, *banners* e animações em *Flash*, podem ser aproveitados do *cache*, diminuindo o tempo total de carregamento.

4. Dependendo da configuração, o *proxy* pode apenas acelerar o acesso às páginas ou servir como um verdadeiro *cache* de arquivos, armazenando atualizações do Windows Update, *downloads* diversos e pacotes instalados através do apt-get, por exemplo. Ao invés de ter que baixar o Service Pack XYZ do Windows XP ou o OpenOffice nos 40 micros da rede, o usuário vai preci-

sar baixar apenas no primeiro, pois os outros 39 poderão baixar a partir do *cache* do Squid.

5. Registra os acessos dos usuários em arquivos. O Administrador de rede poderá visualizar os acessos posteriormente, usando os relatórios do Squid Analysis Report Generator, assim ele saberá quem acessou quais páginas e em que horários, que auxilia a incrementar as listas de bloqueios.

6. Outra função interessante do *proxy* são as políticas de controle denominadas ACL (AccesControl List). Como as requisições são feitas para ele, torna bastante útil do ponto de vista da empresa que quer ter controle sobre o que os empregados estão acessando, podendo definir o que eles podem o não acessar durante o expediente. Se por exemplo, existirem regras que impeçam a passagem de qualquer endereço *WEB* que contenha a palavra "sexo", esse pedido será descartado.

Seguem os parâmetros e as descrições do arquivo `squid.conf`, ilustrado na Figura 3.3:

http_port: Especifica em qual porta o daemon deve aguardar por conexões, dentre os valores mais utilizados estão as portas 3128 e a 8080.

cache_mem: Diretiva que especifica a quantidade de memória em *MegaBytes* a ser disponibilizada para o Squid.

cache_dir: É composta por quatro valores, conforme Tabela 3.1.

cache_access_log: Especifica a localização do arquivo que deve fazer os resgistros de acessos, muito utilizado pelo Squid Analysis Report Generator para gerar as páginas com as estatísticas de acesso.

cache_log: Especifica o arquivo para registros das informações relativas ao cache de arquivos.

cache_store_log: Registro detalhado de todo objeto armazenado detalhando quais objetos saíram, entraram e quanto tempo estes objetos estiveram armazenados.

cache_effective_user: Especifica o usuário dono dos processos criados pelo Squid.

cache_effective_group: Especifica o grupo dono dos processos criados pelo Squid.

```
1  http_port 3128 transparent
2  cache_mem 128 MB
3  visible_hostname www.ce10.com.br
4  cache_swap_low 90
5  cache_swap_high 95
6  maximum_object_size 4096 KB
7
8  cache_dir ufs /usr/local/squid/cache 500 30 384
9  cache_access_log /usr/local/squid/logs/access.log
10 cache_log /usr/local/squid/logs/cache.log
11 cache_store_log usr/local/squid/logs/store.log
12
13 acl all src 0.0.0.0/0.0.0.0
14 acl manager proto cache_object
15 acl SSL_ports port 443 563
16 acl Safe_ports port 80     # http
17 acl Safe_ports port 21     # ftp
18 acl Safe_ports port 443 563   # https, snews
19 acl Safe_ports port 70     # gopher
20 acl Safe_ports port 210    # wais
21 acl Safe_ports port 1025-65535   # unregistered ports
22 acl Safe_ports port 280    # http-mgmt
23 acl Safe_ports port 488    # gss-http
24 acl Safe_ports port 591    # filemaker
25 acl Safe_ports port 777    # multiling http
26 acl CONNECT method CONNECT
27
28 acl REDEINTERNA src 192.168.0.0/255.255.255.0
29 http_access allow REDEINTERNA
30 http_access deny all
31 cache_effective_user squid
32 cache_effective_group squid
```

Figura 3.3: Exemplo do arquivo Squid.conf

ACLs: Que são demonstradas na Figura 3.3, onde os usuários da rede 192.168.0.0/24, denominada REDEINTERNA terá seu acesso ao *proxy* liberado no parâmetro "http_access allow REDEINTERNA" . Em seguida, o parâmetro "http_access deny all" que bloqueia todo acesso não liberado anteriormente por uma *ACL*.

3.4 Tipos de Configurações do Squid

O Squid funciona em 3 modos de configurações: convencional, autenticado e em modo transparente, os quais serão descritos nas subseções 3.4.1, 3.4.2, 3.4.3.

Tabela 3.1: Descrição do parâmetro cache_dir no Squid

Valor	Descrição
ufs	Especifica o tipo de armazenamento padrão a ser utilizado
/etc/squid/cache	Indica a pasta onde o Squid armazena os arquivos do *cache*
500	A quantidade de espaço em disco, em MB, que deve ser alocado para *cache*
30	Quantos diretórios de primeiro nível devem ser criados
384	Quantos diretórios de segundo nível devem ser criados

3.4.1 *P*roxy Convencional

Com o uso do *proxy* em modo convencional, é necessário configurar em cada máquina o navegador e todos os outros programa, que forem acessar a Internet conforme ilustra a Figura 3.4 . Esta é uma tarefa tediosa e que acaba gerando bastante trabalho, pois toda vez que um micro novo for colocado na rede ou for preciso reinstalar o sistema, será preciso fazer a configuração novamente (MORIMOTO, 2008). Em contrapartida, é nesse modelo que ele funciona em modo autenticado podendo ser uma alternativa a mais para segurança da instituição.

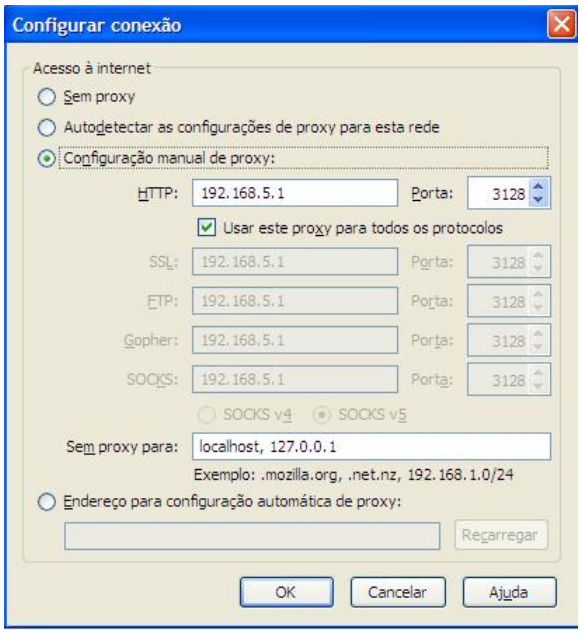


Figura 3.4: Configuração do Proxy do Mozilla Firefox.

3.4.2 *P*roxy em Modo Autenticado

O modo autenticado fornece a rede de computadores uma camada a mais de segurança. A Figura 3.5 ilustra a sequência dos processos executados durante a tentativa de acesso a Internet através de um *proxy* configurado para autenticar seus usuários. Primeiro, uma estação de trabalho cujo seu navegador *Web* tenha sido configurado para utilizar *webproxy* como meio de acesso a Internet, terá que apresentar suas credenciais de acesso para obter o acesso desejado. O questionamento se faz geralmente através de um formulário no qual seve ser digitado o nome de usuário e sua respectiva senha.

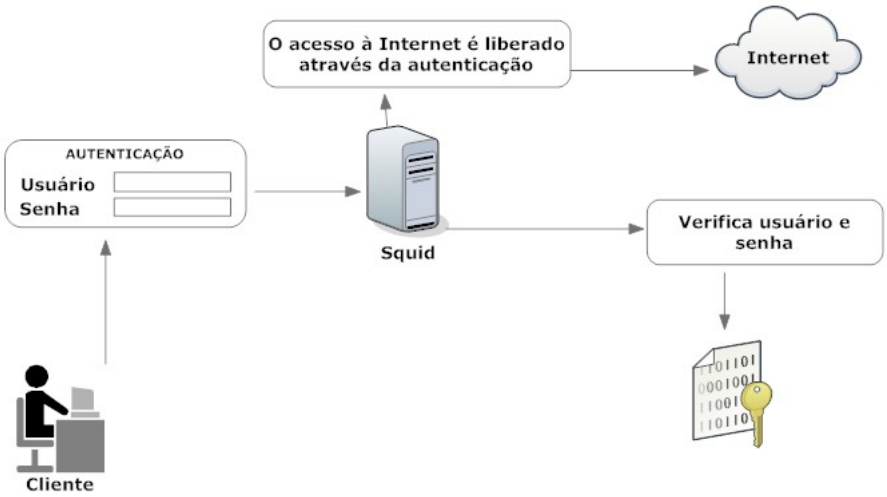


Figura 3.5: Processo de autenticação no *Proxy* Squid.

3.4.3 *P*roxy em Modo Transparente

É possível configurar o Squid e o *firewall* de forma que o servidor *proxy* fique escutando todas as conexões na porta 80. Mesmo que alguém tente desabilitá-lo manualmente nas configurações do navegador, ele continuará sendo usado (MORIMOTO, 2008). Nesse modelo o usuário muitas vezes nem sabe que ele está passando por um *proxy*. Todavia não será necessário visitar localmente as estações de trabalho para configurar seus navegadores Web, que facilita bastante o trabalho do Administrador de Rede. Todo acesso à Internet será forçosamente obtido através do serviço de *proxy*. Para ativar o suporte ao modo transparente é preciso incluir

no início do arquivo squid.conf, para as versões 2.6 em diante, a linha http_port 3128 transparent.

A configuração do Squid é feita através da edição do arquivo `squid.conf`, sua localização pode ser determinado no momento da instalação. A Figura 3.3 ilustra um exemplo de configuração básica desse arquivo. Vale ressaltar que, a ordem interpretada pelo *webproxy* Squid é em ordem de sequência.

3.5 ACL

As ACLs *(Access Control Lists)*, ou Listas de Controle de Acesso, constituem-se na grande flexibilidade e eficiência do Squid, é através delas que podem ser criadas regras para controlar o acesso à Internet. Praticamente todo o processo de controle do Squid é feito com o seu uso. O uso das listas de controle de acesso é a parte mais importante da configuração de um servidor *Proxy* Squid, entretanto se mal configuradas podem oferecer resultados opostos, já que além da falsa sensação de segurança não será aproveitada a grande capacidade e funcionalidade do Squid. A declaração da ACL definem a combinação de permissão *(allow)* e negação *(deny)* de acesso (http_access), implementando a política e controle de acesso à *Web*.

3.5.1 Sintaxe das Regras de Acesso e Diretivas de Controle

- `acl rede_1 src 192.168.2.0/24` : A ACL cria a lista de acesso chamada rede_1 do tipo src (origem) onde é definido a origem da requisição de acesso aos computadores da rede 192.168.2.0.

- `http_access deny rede_1` : Define que os elementos contidos na lista de acesso rede_1 terá seu acesso bloqueado.

3.5.2 Classes ou Tipos de Regras de Acesso

- **SRC**: Classe de acesso que se baseia no endereço *IP* da origem da requisição. Ou seja, a regra se baseará no endereço do cliente que solicita a requisição de acesso.
  Exemplo: `acl REDEINTERNA src 192.168.0.0/255.255.255.0`.

- **DST**: Classe de acesso semelhante ao tipo anterior, mas está relacionada ao endereço de destino. Exemplo: `acl NEGAR dst 200.175.44.1/255.255.255.255`.

- **SCRDOMAIN**: Classe de acesso que se baseia no domínio *DNS* (Domain Name System) do computador cliente que solicita a requisição de acesso.

- **DSTDOMAIN**: Classe de acesso semelhante ao tipo anterior, mas está relacionada ao servidor de destino da solicitação gerada pelo cliente.
- **TIME**: Classe onde é possível determinar o horário, dia da semana em que as requisições serão liberadas ou não o acesso a Internet.
- **URL_Regex**: Classe de acesso percorre a *URL* (Uniform Resource) a procura da expressão regular específica. É importante resaltar que a expressão é case-sensitive, para que seja case-insensitive deve ser usada a opção -i, na qual é possível bloquear ou liberar as requisições solicitados pelo cliente.
- **Port**: Classe de acesso onde é possível controlar o acesso baseado na porta de destino do servidor solicitado pelo cliente .
- **Proto**: Classe de acesso na qual é possível especificar o protocolo utilizado na conexão.
  Exemplo: `acl FTP proto FTP`
- **ARP**: Classe de acesso responsável por liberar ou bloquear o acesso baseado no endereço físico da placa de rede *MAC Address* (Media Access Control address).
- **Proxy_Auth**: Classe de acesso na qual permite a autenticação de usuários através de suas credenciais, que são geralmente o nome de usuário e sua respectiva senha.
- **HTTP_Access**: Permite ou nega acesso ao serviço *HTTP* (Hyper Text Transfer Protocol) baseado na lista de acesso.

Segundo (SILVA, 2009) alguns tipos de ACLs do Squid são notáveis consumidores de CPU. Dentre eles podem ser destacados os tipos *url_regex, urlpath_regex, srcdom_regex e dstdom_regex*. Um mínimo de ACLs deve ser configurado em implementações do Squid como aceleradores HTTP em Web *sites* muito requisitados.

3.6 Squid Analysis Report Generator

O Squid Analysis Report Generator, ou SARG é uma ferramenta de código livre, interpretador dos registros de acessos do Squid utilizada para auditar o acesso à Internet de seus usuários. Dentre suas qualidades destacam-se sua velocidade, a licença de software livre.

O SARG foi utilizado no presente trabalho para auditar os acessos dos usuários e incrementar melhor as listas de bloqueios do DansGuardian, e até mesmo para

tomar as medidas cabíveis em casos de abusos. Conforme (MORIMOTO, 2008): os filtros de conteúdos nunca são completamente eficazes, ele sempre bloqueiam algumas páginas úteis e deixam passar muitas páginas impróprias.

3.7 DansGuardian

DansGuardian é uma premiada ferramenta *open-source* para filtro de conteúdo *web*, desenvolvida para trabalhar com *proxy*, este permite obter uma grande flexibilidade pois oferece recursos como comparação de palavras, e também, expressões regulares (RICCI; MENDONÇA, 2006).

O DansGuardian possui um conjunto de regras contendo palavras, frases, quase todos tipos de páginas indesejadas que são scaneadas em tempo real, além de vários tipos de extensões de arquivos que infectados, trazem riscos ao computador. As listas do DansGuardian podem ser baixadas no *site* `http://urlblacklist.com`. Ela é uma lista comercial, que inclui mais de 2 milhões de endereços, e é atualizada regulamente.

Existem grupos destinados a manter listas com endereços de páginas de cassinos e jogos, páginas pornográficas e páginas ilícitas em geral, que frequentemente são atualizadas. Essas são montadas através da combinação dos esforços de muitas pessoas de vários paises.

Há lista completamente livre e utilizável para qualquer fim como por exemplo o MESD blacklists. Ela tem pouco mais de 1 milhão de endereços e pode ser baixada no endereço `http://squidguard.mesd.k12.or.us/blacklists.tgz`.

Outra opção é a lista Shalla Secure Services. É uma lista livre para uso pessoal ou não-comercial, e possui 1.7 milhões de endereços cadastrado, que formam um arquivo compactado de 9,27 MB. O uso comercial é permitido, desde que seja preenchido o contrato de uso, sem custo. A lista Shalla Secure Services pode ser baixada no endereço `http://www.shallalist.de/Downloads/shallalist.tar.gz`.

Portanto, com todos esses recursos associada ao Squid, é possível implementar políticas de acesso personalizadas para se adequar a esse cenário, propiciando uma grande economia de banda, pois os *sites* e arquivos que se encontram nas regras do DansGuardian não chegam a ser acessados.

O arquivo principal do DansGuardian fica localizado em /etc/dansguardian/-dansguardian.conf, que é definido pelos seguintes parâmetros:

***language*:** define a língua em que as mensagens de acessos bloqueadas serão mostradas aos clientes.

***loglocatio*:** localização do arquivo de registros de acessos do DansGuardian, onde ficam armazenadas os endereços de páginas bloqueadas.

***filterport*:** porta onde o DansGuardian fica ativo, que deve ser utilizado sempre a porta diferente do Squid.

***proxyip*:** define o endereço IP do servidor *proxy* que será usado.

***proxyport*:** especifica a porta do protocolo *TCP - Transmission Control Protocol* onde o Squid está ativo.

***filtergroupslist*:** serve para definir a quantidade de grupos de usuários, que serão usado no DansGuardian.

Para configuração dos grupos de usuários é necessário modificar os arquivos: `dansguardianf1.conf`, `dansguardianf2.conf`, `dansguardianf3.conf`, etc. A referência feita a estes arquivos será por meio da utilização do vocábulo dansguardianf(x).conf. Através desses arquivos foi possível definir o perfil de cada grupo, como nos exemplos abaixo:

***dansguardianf1.conf*:** pode ser configurado para bloquear tanto os *downloads* de arquivos, quanto conteúdo de páginas impróprias.

***dansguardianf2.conf* :** endereços IPs ou Grupos de Usuários que terão acesso livre para conteúdo.

***dansguardianf3.conf* :** endereços IPs ou Grupos de usuários com *downloads* de arquivos liberados.

Segue abaixo uma breve descrição de outros arquivos disponibilizados pelo DansGuardian que ficam situados no seu diretório padrão `\etc\dansguardian\lists` :

***exceptioniplist*:** Arquivo de configuração responsável por icar os endereços IPs da rede que não serão filtrados.

***exceptionphraselist*:** Lista de frases que serão exceção, ou seja, se aparecerem no conteúdo de uma página devem ser ignoradas.

***exceptionuserlist*:** Lista de usuários que não serão filtrados.

***exceptionsitelist*:** *sites* Liberados.

***exceptionurllist*:** Listas de URLs que serão liberadas.

***bannediplist*:** Lista de endereços IPs da rede sem acesso à Internet.

***banneduserlist*:** Usuários da rede que serão bloqueados

***bannedregexpurllist*:** Lista de expressões regulares bloqueadas.

***bannedurllist*:** Ao contrário do arquivo `exceptionurllist`, nesse constam as URls que serão bloqueadas.

***bannedsitelis*:** Lista de *sites* que serão bloqueados

***bannedmimetypelist*:** Alista de *MIME-Types*, os quais serão bloqueados para todo e qualquer acesso à Internet.

***bannedextensionlist*:** Lista de extensões de arquivos bloqueados.

***bannedphraselist*:** Lista de frases banidas dentro da página

***weightedphraselist*:** Lista de frases/palavras e seus "pesos" (os pesos podem ser positivos ou negativos).

O DansGuardian possui um método de ponderação de palavras que foi muito relevante nesse projeto. Ele faz uma "leitura"no conteúdo da página e verifica em suas listas a existência desses conteúdos lidos. Esses arquivos ficam armazenados na pasta `/etc/dansguardian/phraselist`, a qual possui vários gêneros (games, pornografia, *malware, chat*, etc) de conteúdo impróprio, com frases e palavras e suas respectivas pontuações.

Cada palavra ruim soma-se um certo número de pontos, como por exemplo: Quando o usuário tenta abrir uma página, que nessa contenha a palavra "sexo" é somado apenas 10 pontos, enquanto a palavra "sexo livre" soma-se 100 pontos. As palavras boas como "biologia", por outro lado subtraem pontos. Ou seja se o usuário estiver acessando conteúdo com frases de sexo e que tenha "sexo livre", ele será bloqueado, mas se for um aluno consultando um assunto de biologia que contenha a palavra "sexo" ele terá 10 pontos contra ele, só que nessa mesma página poderá ter a palavra "biologia", na qual ele terá 100 pontos a favor dele com saldo de 90 pontos para continuar navegando nessa mesma página. O limite tolerado de cada grupo de usuários é definido na opção *naughtynesslimit*, do arquivo `dansguardianf(x).conf`, mencionado anteriormente.

A partir do estudo realizado, a Figura 3.7 ilustra o processo, do projeto aqui proposto, no qual o DansGuardian receberá as requisições do navegador do usuário para navegar na Internet, esse checa a requisição de acordo com o nome de

usuário, endereço IP de origem e verifica o endereço a ser acessado se encontra em uma das listas de domínios, *URLs, IPs*, tipos de arquivos proibidos, etc. Caso esteja em alguma dessas listas, o cliente receberá a mensagem de acesso negado. Do contrário a requisição é enviada para o Squid e o acesso é realizado. Essas regras dificultaram o acesso via Internet de arquivos maliciosos, horas perdidas por parte dos funcionários com conteúdo impróprio e desvio de atenção com *sites* de conteúdos que não tinham nada a acrescentar a suas funções, tornando-os mais produtivos, e consequentemente a economia de banda.

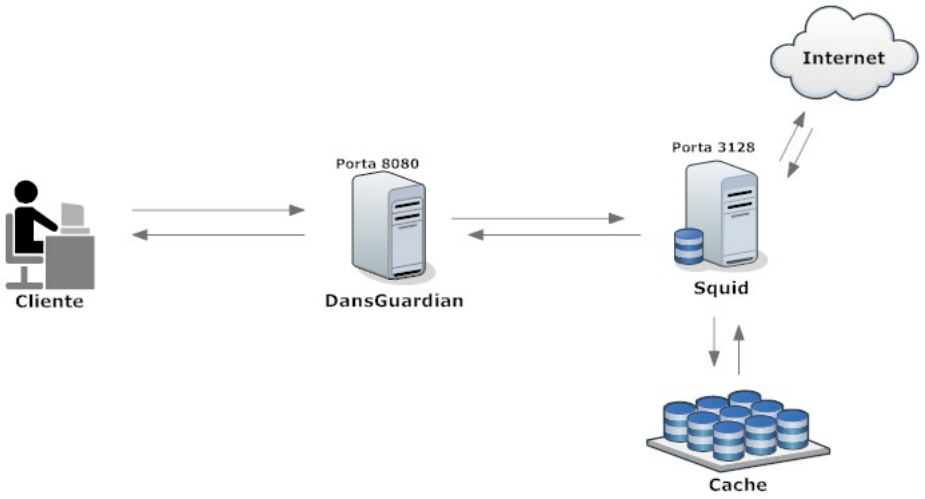


Figura 3.6: Fluxo de requisição do usuário passando pelo filtro de conteúdo

3.8 QoS - Qualidade de serviço

A filosofia de melhor esforço, corresponde ao comportamento padrão aplicado aos pacotes trafegados. Na rede de computadores cada cliente compartilha a largura da banda com os outros. A qualidade desses serviços ou QoS refere-se ao desempenho de uma rede relativa às necessidades das aplicações, como VoIP, correio eletrônico e download de arquivos.

O Hierarchical Token Bucket ou HTB é um mecanismo de escalonamento que foi criado por Martin Devera, sendo atualmente considerado o sucessor do Class-Based Queueing ou CBQ, que segundo (DEVIK, 2002) esse mecanismo é bastante complexo e não ajuda a otimizar em situações típicas.

O HTB implementa um escalonador classfull para o controle de tráfego, fornece métodos para controlar a largura de banda para cada classe trafegada. Os dados empacotados são encaminhados da melhor forma a proporcionar garantias "largura de banda" para determinado tipo de tráfego e com isso melhorar o desempenho da rede.

Portanto, embasado nas ideias de autores relacionados ao assunto foram mostradas todas essas ferramentas que serviram de base para a implantação desse projeto, sendo que a instalação e a configuração das mesmas serão evidenciadas no desenvolvimento do presente trabalho.

Capítulo 4

A INSTALAÇÃO E CONFIGURAÇÃO DAS FERRAMENTAS

Neste capítulo será mostrada a instalação e configuração das ferramentas apresentadas no Capítulo 3, que foram necessárias para a implantação desse projeto.

4.1 Instalação do Sistema Operacional

O Sistema Operacional adotado foi o Slackware na versão 12.2.0 x86_64, conforme foi justificado na seção 3.1. Para a implantação desse projeto a instituição adquiriu um servidor rack da HP-Intel modelo Proliant DL 120 G5, processador Intel Quad-Core Xeon X321, 6 GB de Memória e 2 Discos rígidos de 250GB montados de acordo com a Figura 4.1.

Segundo (FLICKENGER, 2006), o *proxy* terá melhor desempenho quando são criados diversos diretórios pequenos separados por discos, que usar um *cache* muito grande em apenas um disco. Com base no contexto acima, verifica-se que com mais de um disco, o processo de acesso ao *cache* será mais balanceado, podendo com isso ganhar em tempo de acesso.

```
/dev/sda5  swap            swap   defaults                    0   0
/dev/sda1  /               ext4   defaults                    1   1
/dev/sda2  /mnt/cache      ext4   noatime,async,noexec,nosuid 1   0
/dev/sdb2  /mnt/cache1     ext4   noatime,async,noexec,nosuid 1   0
/dev/sda4  /mnt/windows    ext4   defaults                    1   2
/dev/sdb1  /mnt/alexbackup ext4   defaults                    1   2
/dev/cdrom /mnt/cdrom      auto   noauto,owner,user           0   0
/dev/fd0   /mnt/floppy     auto   noauto,owner                0   0
```

Figura 4.1: Arquivo fstab

4.2 Compilação, Instalação e Configuração do Squid

Na instalação do Squid foi utilizado a versão *2.6.STABLE13*, em modo transparente não autenticado, na qual os benefícios citados na subseção 3.4.3, na qual foi realizada através dos seguintes comandos:

#./configure –prefix=/etc/squid –enable-err-Portuguese=lang

#make

#make install

Após a instalação, foram feitas as alterações no arquivo de configuração do Squid que fica localizado no diretório /etc/squid/squid.conf, como ilustra a figura 4.2.

Onde:

- **Linha 2** : para a configuração da quantidade de memória RAM dedicada ao Squid é feita adicionando a opção “cache_mem”.
  Segundo (VESPERMAN, 2003), o Squid precisará de mais memória para indexá-lo conforme o tamanho do Cache é aumentado. No entanto há um cálculo de memória com relação ao espaço de disco utilizado pelo Squid que funciona da seguinte forma:

  Divide-se o tamanho do Cache desejado por 13 Kbytes, e multiplica-se o resultado por 130 bytes. Acrescenta-se o tamanho de cache_mem, e mais 2.3

Mbytes para arquivos executáveis, bibliotecas, e outras cargas. Sendo assim, para a configuração do Squid referente a figura 4.2, o servidor precisará de 308,50 Mbytes de memória RAM disponíveis para o Squid .

- **Linha 6**: a diretiva **visible_hostname** define o nome do servidor, no caso www.ce10.com.br.

- **Linha 8**: a diretiva **maximum_object_size** com 512 MB de memória. Onde determina o tamanho máximo dos arquivos que serão guardados no*cache* feito na memória RAM, uma vez que, segundo (MORIMOTO, 2008) o *cache* na memória RAM, é muito mais rápido. Por se tratar de uma quantidade de memória muito limitada, a mesma ficará mais reservada para páginas *Web*, figuras e arquivos pequenos em gerais.

- **as linhas 11 à 12**: a ACL "ips_liberados" foi utilizado em algumas máquinas na rede que precisavam estar com seu acesso com a Internet totalmente liberada. A opção "time", da ACL "alex_libHor", foi possível fazer a liberação de acesso a Internet por horário, conforme as linhas 15 à 19.

- **Linha 22**: a diretiva **cache_dir ufs /usr/local/squid/cache 5000 30 384**, define o caminho do *cache* do Squid com a opção "/usr/local/squid/cache" , Após isso, temos 5000, que se refere ao tamanho máximo que poderá ser utilizado pelo Squid para armazenar arquivos. ou seja, 5000 Mbytes de espaço em disco. Logo após, foi definido mais 2 números. que significa : o primeiro , 30, representa os diretórios de primeiro nível que o Squid terá, ou seja , no diretório /usr/local/squid/cache, poderão ser criados 30 diretórios; o segundo número, 384, refere-se aos diretórios de segundo nível, que nada mais são do que a quantidade de subdiretórios que os primeiros 30 podem ter.

- **Linha 23**: todas as requisições e atividades dos clientes controlados pelo Squid ficarão gravadas no arquivo /usr/local/squid/logs/access.log.

- **As linhas com as ACLs all src 0.0.0.0/0.0.0.0 e http_access allow 'rede'** : essas linhas criam uma política de acesso chamada *all* (todos). Elas permitem que qualquer uma máquina que estiver dentro desta lista use o *proxy*, sem limitações

- **as linhas 29 à 39**: com as diretivas "acl SSL_ports port" e a "Safe_ports port" são responsáveis por limitar as portas que podem ser usadas através do *proxy*.

- **as linhas 45 à 47**: definem as redes denominadas “gerencia”, “lab” e “localhost” e suas respectivas faixas de *IPs* que usarão o *proxy* para se conectar com a Internet. As linhas 56, 57 e 58 estão as diretivas responsáveis pelas as suas liberações http_access allow “nomeDaRede”.

- **a linha 49**: define o arquivo onde se encontra os endereços da *Web* (ACL liberar_url) que serão liberados para a rede.

- **as linhas 50 e 51**: definem os arquivos onde ficarão os endereços da *Web* (ACL bloquear) e os tipos de arquivos (ACL download) a serem bloqueados para o acesso a Internet.

- **a linhas 59** a diretiva http_access deny all bloqueia todos os clientes que não se enquadram nas regras anteriores. Especificamente as linhas 55 a 58 .

É importante destacar que a ordem das ACLs e de extrema importância, uma vez que o Squid interpreta as regras na ordem que são colocadas no arquivo. Se for definido que o micro X acesse o *proxy*, ele poderá acessar, mesmo que uma regra mais abaixo do arquivo defina o contrário.

Exemplo:

```
acl rede_local src 192.168.200.0/24

http_access allow rede_local

http_access deny rede_local
```

Nesse exemplo os computadores da rede local denominada na ACL “rede_local” continuriam acessando a Internet, pois a regra que permite o acesso vem antes da que proíbe.

```
1  http_port 3128 transparent
2  cache_mem 256 MB
3  visible_hostname www.ce10.com.br
4  cache_swap_low 90
5  cache_swap_high 95
6  maximum_object_size 512 MB
7
8  cache_dir ufs /usr/local/squid/cache 5000 30 384
9  cache_access_log /usr/local/squid/logs/access.log
10 cache_log /usr/local/squid/logs/cache.log
11 cache_store_log usr/local/squid/logs/store.log
12
13 acl all src 0.0.0.0/0.0.0.0
14 acl manager proto cache_object
15 acl SSL_ports port 443 563
16 acl Safe_ports port 80     # http
17 acl Safe_ports port 21     # ftp
18 acl Safe_ports port 443 563   # https, snews
19 acl Safe_ports port 70     # gopher
20 acl Safe_ports port 210    # wais
21 acl Safe_ports port 1025-65535   # unregistered ports
22 acl Safe_ports port 280    # http-mgmt
23 acl Safe_ports port 488    # gss-http
24 acl Safe_ports port 591    # filemaker
25 acl Safe_ports port 777    # multiling http
26 acl CONNECT method CONNECT
27
28 acl REDEINTERNA src 192.168.0.0/255.255.255.0
29 http_access allow REDEINTERNA
30 http_access deny all
31 cache_effective_user squid
32 cache_effective_group squid
```

Figura 4.2: Squid.conf somente para cache, e controle de endereço IP

4.3 Instalação e Configuração do DansGuardian

A versão do DansGuardian utilizado foi o 2.10.1.1-19. Optou-se em fazer a instalação pelo pacote tgz, com o comando :

```
#installpkg dansguardian-2.10.1.1-19.1.x86_64.tgz.
```

Segundo (MORIMOTO, 2008) os pacotes tgz são basicamente programas pré-compilados junto com um script de instalação especificando os diretórios para onde os arquivos devem ser copiados, providenciando a criação dos arquivos de

configuração necessários, entre outras tarefas.

A Figura 4.3 ilustra a estrutura de pastas do DansGuardian após a instalação e configuração.

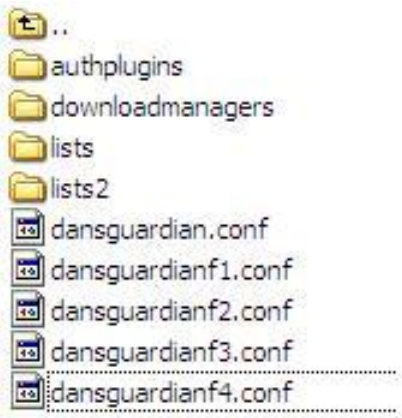


Figura 4.3: Estrutura de pastas do DansGuardian.

Após a instalação do DansGuardian, foi necessário editar alguns arquivos e diretórios, para melhor se adequar a necessidade da Instituição.

O arquivo `dansguardian.conf`, foi editado com as seguinte opções:

- **language = portuguese** : conforme já foi explicado na seção 3.7, esse parâmetro define a língua em que as mensagens de acesso bloqueado serão exibidas aos usuários.
- **logfileformat = 3 :** o número 3 indica o formato do arquivo de registro de acesso usado pelo Squid, onde serão tratados no *Squid Analysis Report Generator*.
- **filtergroups = 4 :** onde será possível definir a quantidade de grupos citado na seção 3.7 do capítulo 3.
- **ipgroups** : para utilização do Squid em modo não autenticado, torna-se necessário fazer a administração pelo endereço *IP* do usuário. Para esse tipo de controle foi preciso configurar o arquivo ipgroups do DansGuardian , localizado no diretório /etc/dansguardian/listspauthplugins, que pode definir o

tipo de filtro em que o endereço *IP* será tratado. Esse arquivo foi configurado conforme Figura 4.4.

Os arquivos `dansguardianf(x).conf` conforme foram apresentados na seção 3.7, foi editada a opção **naughtynesslimit** que define o índice máximo tolerado conforme citado na sessão 3.7 na qual foi ajustado para cada grupo de usuários, conforme são mostradas nas Figuras 4.5, 4.6, 4.7.

Foi preciso criar uma cópia da pasta **lists**, na qual foi nomeada como **lists2**, conforme Figura 4.3. No entanto foi preciso limpar todo o conteúdo dos arquivos contidos na mesma, para que toda configuração apontada para ela fossem liberados conforme Figura 4.5.

```
-------------------------- Legenda ------------------------------
#filter1 Bloqueio total (Com índice de tolerância bem baixo, onde
será ultilizado no laboratório
 de informática)
#filter2  download liberados
#filter3 todo o acesso a conteúdo liberado
#filter4 liberação total
-----------------------------------------------------------------

192.168.10.0/255.255.255.0 =filter1
#
192.168.15.2  =filter2
192.168.15.5  =filter4
192.168.15.25 =filter1
192.168.15.45 =filter3
```

Figura 4.4: Exemplo do arquivo ipgroups

As Figuras 4.5, 4.6, 4.7, ilustra os parâmetros principais definidos nos arquivos dansguardianf(x).conf citado na seção 3.7.

4.4 Instalação e Configuração do Analysis Report Generator

O *Analysis Report Generator* foi utilizado a versão 2.2.1-1.fc3.rf.x86_64, instalada através do pacote tgz. O arquivo principal de configuração dessa ferramenta

```
groupmode = 1
naughtynesslimit = 75
# Filter group name
# Content filtering files location
bannedphraselist = '/etc/dansguardian/lists/bannedphraselist'
weightedphraselist = '/etc/dansguardian/lists/weightedphraselist'
exceptionphraselist = '/etc/dansguardian/lists/exceptionphraselist'
bannedsitelist = '/etc/dansguardian/lists/bannedsitelist'
greysitelist = '/etc/dansguardian/lists/greysitelist'
exceptionsitelist = '/etc/dansguardian/lists/exceptionsitelist'
bannedurllist = '/etc/dansguardian/lists/bannedurllist'
greyurllist = '/etc/dansguardian/lists/greyurllist'
exceptionurllist = '/etc/dansguardian/lists/exceptionurllist'
exceptionregexpurllist = '/etc/dansguardian/lists/exceptionregexpurllist'
bannedregexpurllist = '/etc/dansguardian/lists/bannedregexpurllist'
picsfile = '/etc/dansguardian/lists/pics'
contentregexplist = '/etc/dansguardian/lists/contentregexplist'
urlregexplist = '/etc/dansguardian/lists/urlregexplist'


exceptionextensionlist = '/etc/dansguardian/lists/exceptionextensionlist'
exceptionmimetypelist = '/etc/dansguardian/lists/exceptionmimetypelist'
bannedextensionlist = '/etc/dansguardian/lists/bannedextensionlist'
bannedmimetypelist = '/etc/dansguardian/lists/bannedmimetypelist'
exceptionfilesitelist = '/etc/dansguardian/lists/exceptionfilesitelist'
exceptionfileurllist = '/etc/dansguardian/lists/exceptionfileurllist''
```

Figura 4.5: Principais parâmetros para definição do perfil grupo nº1, que será utilizado para o Laboratório de Informática

é o sarg.conf, localizado no diretório /etc/sarg/. Os seus relatórios não rodam como um *deamon*, os mesmos são construídos no momento da sua execução.

É possível customizar os relatórios com informações relevantes como por exemplo: *sites* mais visitados, usuários que visitaram determinados *sites*, etc (SAMORUKOV, 2010).

Para a geração dos relatórios foi utilizado o comando :

```
groupname = '2'
naughtynesslimit = 250
# Content filtering files location
bannedphraselist = '/etc/dansguardian/lists/bannedphraselist'
weightedphraselist = '/etc/dansguardian/lists/weightedphraselist'
exceptionphraselist = '/etc/dansguardian/lists/exceptionphraselist'
bannedsitelist = '/etc/dansguardian/lists/bannedsitelist'
greysitelist = '/etc/dansguardian/lists/greysitelist'
exceptionsitelist = '/etc/dansguardian/lists/exceptionsitelist'
bannedurllist = '/etc/dansguardian/lists/bannedurllist'
greyurllist = '/etc/dansguardian/lists/greyurllist'
exceptionurllist = '/etc/dansguardian/lists/exceptionurllist'
exceptionregexpurllist = '/etc/dansguardian/lists/exceptionregexpurllist'
bannedregexpurllist = '/etc/dansguardian/lists/bannedregexpurllist'
picsfile = '/etc/dansguardian/lists/pics'
contentregexplist = '/etc/dansguardian/lists/contentregexplist'
urlregexplist = '/etc/dansguardian/lists/urlregexplist'

exceptionextensionlist = '/etc/dansguardian/lists2/exceptionextensionlist'
exceptionmimetypelist = '/etc/dansguardian/lists2/exceptionmimetypelist'
bannedextensionlist = '/etc/dansguardian/lists2/bannedextensionlist'
bannedmimetypelist = '/etc/dansguardian/lists2/bannedmimetypelist'
headerregexplist = '/etc/dansguardian/lists2/headerregexplist'
bannedregexpheaderlist = '/etc/dansguardian/lists2/bannedregexpheaderlist'
```

Figura 4.6: Principais parâmetros para definição do perfil grupo nº2, os *downloads* serão liberados

/usr/bin/sarg -f /etc/sarg/sarg.conf -d 26/03/2010 -p

Onde:

- **/usr/bin/sarg**: Localização do sistema.
- **-f :** Esse parâmetro indica a localização do arquivo sarg.conf.
- **-d:** Permite definir a data de geração dos relatórios.
- **26/03/2010 :** Período para geração dos relatórios.
- **-p:** Resolve o endereço *IP* do usuário de acordo com o arquivo hosts

```
groupname = '3'
naughtynesslimit = 250
# Content filtering files location
bannedphraselist = '/etc/dansguardian/lists2/bannedphraselist'
weightedphraselist = '/etc/dansguardian/lists2/weightedphraselist'
exceptionphraselist = '/etc/dansguardian/lists2/exceptionphraselist'
bannedsitelist = '/etc/dansguardian/lists2/bannedsitelist'
greysitelist = '/etc/dansguardian/lists2/greysitelist'
exceptionsitelist = '/etc/dansguardian/lists2/exceptionsitelist'
bannedurllist = '/etc/dansguardian/lists2/bannedurllist'
greyurllist = '/etc/dansguardian/lists2/greyurllist'
exceptionurllist = '/etc/dansguardian/lists2/exceptionurllist'
exceptionregexpurllist = '/etc/dansguardian/lists2/exceptionregexpurllist'
bannedregexpurllist = '/etc/dansguardian/lists2/bannedregexpurllist'
picsfile = '/etc/dansguardian/lists2/pics'
contentregexplist = '/etc/dansguardian/lists2/contentregexplist'
urlregexplist = '/etc/dansguardian/lists2/urlregexplist'

exceptionextensionlist = '/etc/dansguardian/lists/exceptionextensionlist'
exceptionmimetypelist = '/etc/dansguardian/lists/exceptionmimetypelist'
bannedextensionlist = '/etc/dansguardian/lists/bannedextensionlist'
bannedmimetypelist = '/etc/dansguardian/lists/bannedmimetypelist'
headerregexplist = '/etc/dansguardian/lists/headerregexplist'
```

Figura 4.7: Principais parâmetros para definição do perfil grupo nº3, somente os conteúdos serão liberados

Foi necessário modificar o arquivo sarg.conf, para que o relatório apresentasse o nome do usuário e não seu endereço *IP*, conforme ilustra a figura 4.9. Nesse caso, alterou-se os parâmetros:

- **user_ip no:** Mostra o endereço IP do usuário, quando o Squid não está funcionando em modo autenticado.

- **usertab /etc/sarg/hosts:** Exibe no relatório o nome do usuário e não o endereço IP, que é resolvido pelo arquivo *hosts*, onde foi inserido os endereços *IPs* da rede e com seus respectivos nomes, conforme é ilustrado na conforme Figura 4.8.

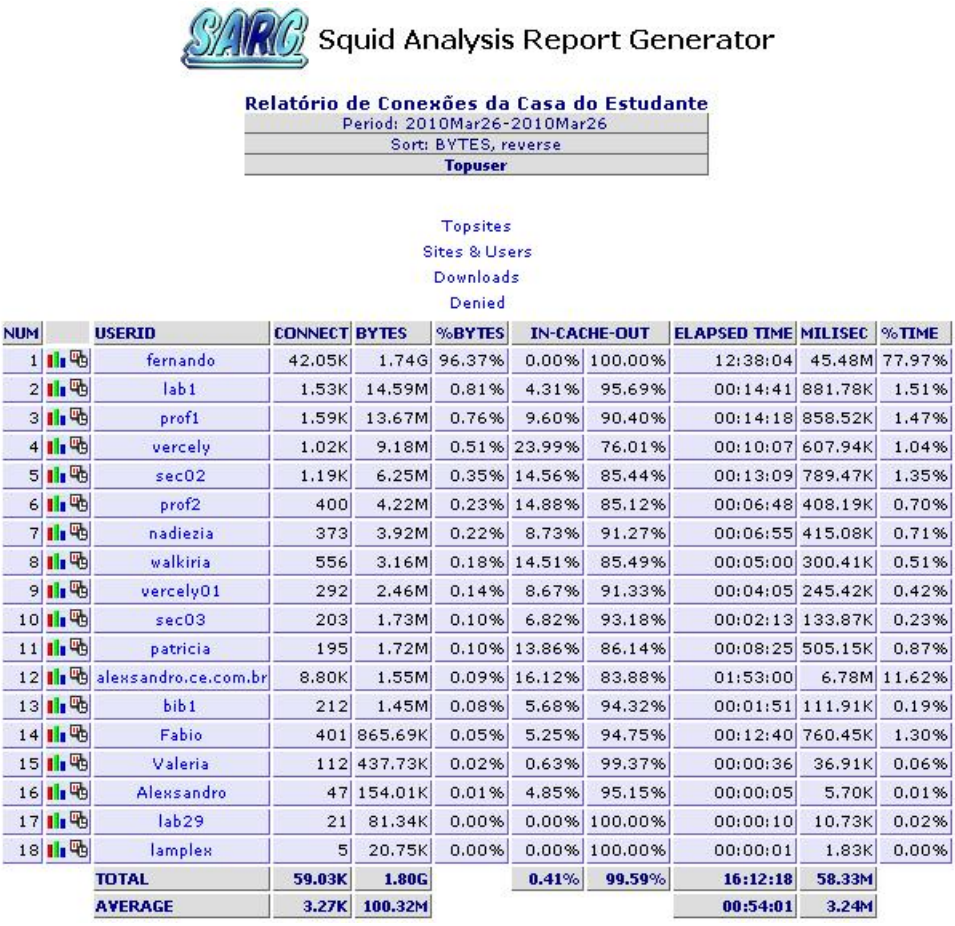
```
192.168.15.5 Alexsandro
192.168.15.6 Iolanda
192.168.15.12 Patricia
192.168.15.30 Fabio
192.168.15.35 Iolanda
192.168.15.40 Daniela
192.168.15.41 Vilany
192.168.15.50 Isabella
```

Figura 4.8: Exemplo do arquivo *hosts*

4.5 *IPTABLES*

Para usar o *proxy* transparente foi preciso compartilhar a conexão com o servidor via *NAT*, na qual o *proxy* intercepta os acessos na porta 80, obrigando tudo a passar pelas suas regras de controle de acesso. O comando abaixo direciona as requisições recebidas na porta 80 para o DansGuardian.

#iptables -t nat -A PREROUTING -i eth1 -p tcp –dport 80 -j REDIRECT –to-port 8080



NUM		USERID	CONNECT	BYTES	%BYTES	IN-CACHE-OUT		ELAPSED TIME	MILISEC	%TIME
1		fernando	42.05K	1.74G	96.37%	0.00%	100.00%	12:38:04	45.48M	77.97%
2		lab1	1.53K	14.59M	0.81%	4.31%	95.69%	00:14:41	881.78K	1.51%
3		prof1	1.59K	13.67M	0.76%	9.60%	90.40%	00:14:18	858.52K	1.47%
4		vercely	1.02K	9.18M	0.51%	23.99%	76.01%	00:10:07	607.94K	1.04%
5		sec02	1.19K	6.25M	0.35%	14.56%	85.44%	00:13:09	789.47K	1.35%
6		prof2	400	4.22M	0.23%	14.88%	85.12%	00:06:48	408.19K	0.70%
7		nadiezia	373	3.92M	0.22%	8.73%	91.27%	00:06:55	415.08K	0.71%
8		walkiria	556	3.16M	0.18%	14.51%	85.49%	00:05:00	300.41K	0.51%
9		vercely01	292	2.46M	0.14%	8.67%	91.33%	00:04:05	245.42K	0.42%
10		sec03	203	1.73M	0.10%	6.82%	93.18%	00:02:13	133.87K	0.23%
11		patricia	195	1.72M	0.10%	13.86%	86.14%	00:08:25	505.15K	0.87%
12		alexsandro.ce.com.br	8.80K	1.55M	0.09%	16.12%	83.88%	01:53:00	6.78M	11.62%
13		bib1	212	1.45M	0.08%	5.68%	94.32%	00:01:51	111.91K	0.19%
14		Fabio	401	865.69K	0.05%	5.25%	94.75%	00:12:40	760.45K	1.30%
15		Valeria	112	437.73K	0.02%	0.63%	99.37%	00:00:36	36.91K	0.06%
16		Alexsandro	47	154.01K	0.01%	4.85%	95.15%	00:00:05	5.70K	0.01%
17		lab29	21	81.34K	0.00%	0.00%	100.00%	00:00:10	10.73K	0.02%
18		lamplex	5	20.75K	0.00%	0.00%	100.00%	00:00:01	1.83K	0.00%
		TOTAL	**59.03K**	**1.80G**		**0.41%**	**99.59%**	**16:12:18**	**58.33M**	
		AVERAGE	**3.27K**	**100.32M**				**00:54:01**	**3.24M**	

Figura 4.9: Relatório do *Analysis Report Generator* gerado na rede do Centro Educacional Casa do Estudante

Capítulo 5

PROBLEMAS OCORRIDOS NA IMPLANTAÇÃO E OS RESULTADOS OBTIDOS

5.1 Testes e implantação

A princípio foi instalado somente o Squid como controle de conteúdo e *cache* de arquivos e páginas *Web* como ilustra o Apêndice A.1, bem como, o *HTB-Tools* para controle de banda por departamento, devido ao alto consumo de banda totalmente descontrolada nos horários de funcionamento do laboratório de informática, que reduzia drasticamente a velocidade do *link* para os outros departamentos.

Em relação ao Squid, com o passar do tempo, as listas de bloqueios estavam ficando enormes devido a grande quantidade de endereços proibidos que surgiam e até mesmo serviços de *proxy* para burlar os controles.

No segundo encontro presencial do curso Administração de Rede Linux na Ufla, a ferramenta DansGuardian foi bastante citada tornando-se objeto de estudos para substituição da função do *Squid* no controle de conteúdo.

Segundo (MORIMOTO,2008) a grande diferença entre o Squid e o *DansGuardian*, é que o *Squid* se limita a bloquear páginas contidas nas listas, enquanto o *DansGuardian*, utiliza um filtro adaptativo que avalia o conteúdo da página e decide se ela é uma página imprópria, com base no conteúdo, utilizando para isso um conjunto de regras adaptativas.

Diante desse contexto, foi testado o Squid somente como *cache* de páginas e arquivos, e o DansGuardian como controle de conteúdo.

Nos testes realizados no início da implantação, o *DansGuardian* demonstrou bastante eficiente no controle de conteúdo conforme ilustra no relatório na figura 5.1. As suas regras de controles conforme foi citado na seção 3.7, foram ajustadas para se adequar a realidade daquela instituição, como a opção de grupo naughtynesslimit, e alguns *sites* que estavam sendo bloqueados indevidamente.

Os Apêndices A.2, A.3, A.4, ilustram algumas páginas com informações de acesso negado, na rede do Centro Educacional Casa do Estudante, na qual o DansGuardian retorna ao navegador do usuário quando esse tenta acessar algum conteúdo configurado no DansGuardian como proibido.

Com os filtros mais refinados, o uso de controle de banda para a instituição não estava tento tanta utilidade, pois, o grande consumo era devido aos abusos no uso do *link* para conteúdo impróprio e *downloads* de arquivos por conta de funcionários e alunos.

A Figura 5.1, ilustra os acessos à Internet do usuário, tratados pelo Squid Analysis Report Generator, na rede do Centro Educacional Casa do Estudante.

Vale ressaltar que com o uso do Linux para a implantação do presente trabalho não foram necessários grandes investimentos. Nos testes foi utilizado uma máquina da IBM com processador Intel Pentium II, com 128 MB de memória encontrada no deposito de sucatas onde ficam os equipamentos classificados como obsoletos para a instituição, na qual foi instalada o Slackware na versão 11.0 em modo texto.

Segundo (FILHO, 2004) o Linux oferece uma série de alternativas com muita eficiência e efetividade a baixo custo. Nele encontramos alguns dos servidores mais usados no mundo.

5.2 A Nova Estrutura da Rede

Em maio de 2009 foi feito investimentos em servidores conforme foi citado na sessão 4.1, por conta do ótimo resultado alcançado e credibilidade conquistada por esse projeto, permanecendo com a seguinte estrutura de servidores:

- **1º servidor:** O Slackware e conforme seção 4.1, com os serviços de *Firewall*, *Proxy cache*, Controle de Conteúdo, *VPN (Virtual Private Network)*, e Servidor de e-mail.
- **2º Servidor** : O Slackware 12.0.0, como controlador de domínio e servidor de impressão

Relatório de Conexões da Casa do Estudante

Period: 2010Mar26-2010Mar26
User:
Sort: BYTES, reverse
User Report

ACCESSED SITE	CONNECT	BYTES	%BYTES	IN-CACHE-OUT		ELAPSED TIME	MILISEC	%TIME	
mail.terra.com.br	56	1.56M	49.46%	14.95%	85.05%	00:01:42	102.35K	34.07%	
www.terra.com.br	12	355.73K	11.24%	41.11%	58.89%	00:00:14	14.33K	4.77%	
bbb.globo.com	12	325.11K	10.27%	0.00%	100.00%	00:00:01	1.86K	0.62%	
sup.live.com	3	248.46K	7.85%	0.00%	100.00%	00:00:12	12.83K	4.27%	
by141w.bay141.mail.live.com	34	216.89K	6.85%	0.00%	100.00%	00:00:23	23.98K	7.98%	
br.msn.com	2	152.00K	4.80%	50.00%	50.00%	00:00:02	2.46K	0.82%	
www.globo.com	9	70.37K	2.22%	0.00%	100.00%	00:00:01	1.38K	0.46%	
stf.terra.com.br	2	67.51K	2.13%	0.00%	100.00%	00:00:25	25.65K	8.54%	
login.live.com	6	60.55K	1.91%	0.00%	100.00%	00:00:07	7.08K	2.36%	
ppi.terra.com.br	15	50.40K	1.59%	0.00%	100.00%	00:00:09	9.09K	3.03%	
www.mozilla.com	3	22.93K	0.72%	0.00%	100.00%	00:00:01	1.82K	0.61%	
www.google.com.br	2	16.60K	0.52%	0.00%	100.00%	00:00:00	595	0.20%	
tagmanfe.terra.com.br	49	10.50K	0.33%	28.68%	71.32%	00:00:30	30.70K	10.22%	
www.google.com	4	1.02K	0.03%	0.00%	100.00%	00:00:05	5.11K	1.70%	
workspace.office.live.com	2	892	0.03%	0.00%	100.00%	00:00:02	2.34K	0.78%	
mail.live.com	3	534	0.02%	0.00%	100.00%	00:00:01	1.66K	0.55%	
video.globo.com	81	0	0.00%	0.00%	0.00%	00:00:00	107	0.04%	DENIED
ad.doubleclick.net	52	0	0.00%	0.00%	0.00%	00:00:00	82	0.03%	DENIED
www.google-analytics.com	41	0	0.00%	0.00%	0.00%	00:00:00	38	0.01%	DENIED
ads.globo.com	32	0	0.00%	0.00%	0.00%	00:00:00	27	0.01%	DENIED
terra.112.2o7.net	26	0	0.00%	0.00%	0.00%	00:00:00	105	0.03%	DENIED
stf.terra.com	23	0	0.00%	0.00%	0.00%	00:00:00	35	0.01%	DENIED
gateway.messenger.hotmail.com	22	0	0.00%	0.00%	0.00%	00:00:44	44.39K	14.78%	DENIED
ads1.msn.com	20	0	0.00%	0.00%	0.00%	00:00:00	20	0.01%	DENIED
h.atdmt.com	16	0	0.00%	0.00%	0.00%	00:00:00	22	0.01%	DENIED
www.sqm.microsoft.com	5	0	0.00%	0.00%	0.00%	00:00:08	8.32K	2.77%	DENIED
45ac8e2907d3ab30.users.storage.live.com	4	0	0.00%	0.00%	0.00%	00:00:00	13	0.00%	DENIED
config.messenger.msn.com	3	0	0.00%	0.00%	0.00%	00:00:00	21	0.01%	DENIED
msnportal.112.2o7.net	2	0	0.00%	0.00%	0.00%	00:00:00	5	0.00%	DENIED
loginnet.passport.com	2	0	0.00%	0.00%	0.00%	00:00:01	1.13K	0.38%	
c.atdmt.com	2	0	0.00%	0.00%	0.00%	00:00:00	2	0.00%	DENIED
b0b4dfb6812a3420.users.storage.live.com	2	0	0.00%	0.00%	0.00%	00:00:00	3	0.00%	DENIED
8282f7f122340ed5.users.storage.live.com	1	0	0.00%	0.00%	0.00%	00:00:00	2	0.00%	DENIED
501b79bad286b431.users.storage.live.com	1	0	0.00%	0.00%	0.00%	00:00:00	1	0.00%	DENIED
4f532004ba8b71de.users.storage.live.com	1	0	0.00%	0.00%	0.00%	00:00:00	1	0.00%	DENIED
3e84d91ac1eb5832.users.storage.live.com	1	0	0.00%	0.00%	0.00%	00:00:00	1	0.00%	DENIED

Figura 5.1: Relatório de conexão de um usuário

- **3º Servidor**: Windows 2003 Server com o banco de dados Microsoft SQL Server 2005.

Capítulo 6

CONCLUSÃO E PROPOSTA DE CONTINUIDADE

É inegável que os serviços oferecidos pela Internet sejam imprescindíveis para o bom desenvolvimento num ambiente corporativo. É nesse ponto que a introdução de ferramentas que ofereçam mais segurança, e otimize esse meio de comunicação não seja apenas um assunto de pesquisa e passa a ser uma exigência desse mercado competitivo. Neste sentido, foram apresentadas ao longo deste trabalho a implantação de ferramentas de software livre que pudessem controlar de forma eficazes o acesso à Internet para otimizar a utilização da banda com mais segurança.

Os resultados obtidos palas ferramentas que foram citadas no desenvolvimento do trabalho propiciaram um melhor controle do conteúdo, bem como, mais segurança no que diz respeito a ameaças vindas da Internet e a otimização desse meio de comunicaçao, haja vista que, com a melhoria da segurança, o controle eficaz do conteúdo, e a utilização de serviço de *proxy* cache na rede da Instituição, houve uma grande melhoria no desempenho da Internet com baixo custos devido à utilização exclusiva de softwares livres.

As ferramentas adotadas mostram que o patrimônio de uma empresa na área da Tecnologia é composta por bens que são mais fáceis de mensurar como os servidores, estações de trabalho, impressoras entre outros. E bens que são mais difíceis de avaliar sendo constituídos pela informação no sentido mais amplo, como todo patrimônio intelectual que têm sempre valor superior aos outros. Esses transformam matérias-primas em produtos manufaturados, equilibrando qualidade, custos e receitas,devendo, portanto, ser protegido contra acessos não autorizados e amea-

ças diversas, como os que são ilustrados nos gráficos das Figuras 1.2, 1.3, 1.1.

O trabalho serviu também para mostrar que a visão do profissional enquanto administrador de rede Linux, deve estar voltada não somente para o conhecimento das ferramentas, mas, sobretudo, para que essas atendam as necessidades da empresa de modo que os benefícios alcançados possam superar os investimentos realizados, onde a qualidade e a produtividade da empresa sejam alcançada de forma mais satisfatória.

Como fonte de dados e informações para a contextualização do trabalho, além das disciplinas estudadas no curso em questão e da orientação do professor, foi feito também um estudo embasado nas teorias de alguns autores relacionados ao assunto.

A proposta de continuidade consiste na elaboração de uma Política de Uso da Internet para estabelecer limites, direitos e deveres de utilização desse recurso, estando o funcionário ou aluno sujeito inclusive a sanções pelo descumprimento das diretrizes mencionadas na Política.

O *Proxy* em modo autenticado pelo servidor Samba, onde atualmente já existe o mesmo configurado como PDC, na qual será uma camada extra de segurança para a Instituição.

Também não é possível afirmar que as ferramentas citadas nesse trabalho sejam as melhores alternativas para alcançar maior segurança do conteúdo visitado na Internet, performance desse meio de comunicação e o aumento da produtividade por parte dos funcionários e alunos, mas que elas configuradas corretamente podem ser bastante eficazes para alcançar o objetivo proposto.

Referências Bibliográficas

BRASIL. Lei nº 8.069, de 3 de julho de 1990. CÓDIGO CIVIL, jul. 1990.

BRASIL. Lei nº 10.406, de 10 de janeiro de 2002. CÓDIGO CIVIL, jan. 2002.

CERT.BR. Estatísticas de notificações de spam reportadas ao cert.br. Comitê Gestor da Internet no Brasil. Disponível em: http://www.cert.br/stats/incidentes. Acesso em: feb. 2010.

CERT.BR. Estatísticas de notificações de spam reportadas ao cert.br. Comitê Gestor da Internet no Brasil. Disponível em: http://www.cert.br/stats/spam. Acesso em: feb. 2010.

CERT.BR. Incidentes reportados ao cert.br – janeiro a dezembro de 2009. Comitê Gestor da Internet no Brasil. Disponível em: http://www.cert.br/stats/incidentes/2009-jan-dec/tipos-ataque.html. Acesso em: feb. 2010.

DEVIK, M. D. aka. Htb linux queuing discipline manual - user guide. 2002. Disponível em: http://luxik.cdi.cz/~devik/qos/htb/manual/userg.htmintro. Acesso em: Dez. 2010.

FILHO, A. S. *Domíno Linux: do Básico aos Servidores*. Florianópolis: VisualBooks, 2004. 377 p.

FLICKENGER, R. *How To Accelerate Your Internet*. Trieste: INASP/ICTP, 2006. 313 p. Disponível em: http://bwmo.net/index.html. Acesso em: JUL. 2010.

LIMA, M. B. *Provisão de Serviços Inseguros Usando Filtros de Pacotes com Estados*. São Paulo: UNICAMP – Campinas, 2000. 143 p. Disponível em: http://www.las.ic.unicamp.br/paulo/teses/20000904-MSc-Marcelo.Barbos.Lima-Provisao.de.servicos.inseguros.usando.filtros.de.pacotes.com.estados.pdf. Acesso em: Oct. 2010.

MORIMOTO, C. E. *Servidores Linux, guia prático*. Porto Alegre: Sul Editores, 2008. 735 p.

NAKAMURA, E. T.; GEUS, P. L. de. *Segurança de Redes em Ambientes Cooperativos*. São Paulo: NOVATEC EDITORA, 2007. 488 p.

PINHEIRO, A. C. S. *Uso e configuração do Squid como servidor proxy*. [s.n.], 2010. Disponível em: http://www.squid-cache.org.br/index.php?option=com_content&task=view&id=82&Itemid=27. Acesso em: Mar. 2010.

RIBEIRO, U. *Certificação Linux*. Rio de Janeiro: Axcel Books, 2004. 450 p.

RICCI, B. *Slackware - Guia Prático*. Rio de Janeiro: Editora Ciência Moderna Ltda, 2004. 187 p.

RICCI, B.; MENDONÇA, N. *Squid - Solução Definitiva*. Rio de Janeiro: Editora Ciência Moderna Ltda, 2006. 152 p.

SAMORUKOV, A. *Squid Analysis Report Generator*. [s.n.], 2010. Disponível em: http://sarg.sourceforge.net/sarg.php. Acesso em: Aug 2010.

SILVA, L. do R. B. B. *Aceleração HTTP: Um comparativo de performance entre as soluções Squid e Varnish*. Lavras: Universidade Federal de Lavras - UFLA, 2009. 81 p. Disponível em: http://www.ginux.ufla.br/node/294. Acesso em: Oct. 2010.

VESPERMAN, J. Installing and configuring squid. 2003. Disponível em: http://linuxdevcenter.com/pub/a/linux/2001/07/26/squid.html?page=2. Acesso em: Oct. 2010.

Apêndice A

APÊNDICES

A.1 Squid.conf como controle de conteúdo

```
1  http_port 3128 transparent
2  cache_mem 256 MB
3  hierarchy_stoplist cgi-bin ?
4  acl QUERY urlpath_regex cgi-bin \?
5  no_cache deny QUERY
6  visible_hostname www.ce10.com.br
7
8  maximum_object_size 512 MB
9
10 #libera IP
11 acl ips_liberados src 192.168.0.7 192.168.0.8 192.168.0.14
12 http_access allow ips_liberados
13
14 #libera IP por hora
15 acl alex_libHor src  192.168.0.6
16 acl manha time MTWHF 10:00-12:00
17 acl tarde time MTWHF 14:00-17:00
18 http_access allow alex_libHor manha
19 http_access allow alex_libHor tarde
20
21
22 cache_dir ufs /usr/local/squid/cache 5000 30 384
23 cache_access_log /usr/local/squid/logs/access.log
24 cache_log /usr/local/squid/logs/cache.log
25 cache_store_log usr/local/squid/logs/store.log
26
27 acl all src 0.0.0.0/0.0.0.0
28 acl manager proto cache_object
29 acl SSL_ports port 443 563
```

```
30 acl Safe_ports port 80     # http
31 acl Safe_ports port 21     # ftp
32 acl Safe_ports port 443 563   # https, snews
33 acl Safe_ports port 70     # gopher
34 acl Safe_ports port 210    # wais
35 acl Safe_ports port 1025-65535   # unregistered ports
36 acl Safe_ports port 280    # http-mgmt
37 acl Safe_ports port 488    # gss-http
38 acl Safe_ports port 591    # filemaker
39 acl Safe_ports port 777    # multiling http
40 acl CONNECT method CONNECT
41 #
42
43
44 #
45 acl gerencia src 192.168.0.0/255.255.255.0
46 acl lab src 192.168.0.0/255.255.255.0
47 acl localhost src 127.0.0.0/255.0.0.0
48
49 acl liberar_url url_regex -i "/usr/local/squid/etc/libe_url"
50 acl bloquear    url_regex -i "/usr/local/squid/etc/sites"
51 acl download    url_regex -i "/usr/local/squid/etc/download"
52
53 http_access deny bloquear
54 http_access deny download
55 http_access allow liberar_url
56 http_access allow lab
57 http_access allow gerencia
58 http_access allow localhost
59 http_access deny all
60 always_direct allow all
```

A.2 Página que é retornada ao usuário na tentativa de acessar conteúdo impróprio

A.3 Página que é retornada ao usuário ao tentar baixar arquivos maliciosos

A.4 Página que é retornada ao usuário ao tentar baixar filmes