UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ

CAMPUS LUIZ MENEGHEL

EMERSON LUIZ SILVA

ADMINISTRAÇÃO DE CONTEÚDO WEB UTILIZANDO SERVIDOR PROXY/CACHE – ESTUDO DE CASO

Bandeirantes

2010

EMERSON LUIZ SILVA

ADMINISTRAÇÃO DE CONTEÚDO WEB UTILIZANDO SERVIDOR PROXY/CACHE

Monografia apresentada à Universidade Estadual do Norte do Paraná – *campus* Luiz Meneghel – como requisito parcial para aprovação na disciplina Metodologia Científica do Curso de Sistemas de Informação.

Orientador: Prof. Luiz Fernando Legore do Nascimento

Bandeirantes

2010

EMERSON LUIZ SILVA

ADMINISTRAÇÃO DE CONTEÚDO WEB UTILIZANDO SERVIDOR PROXY/CACHE

Monografia apresentada à Universidade Estadual do Norte do Paraná – *campus* Luiz Meneghel – como requisito parcial para aprovação na disciplina Metodologia Científica do Curso de Sistemas de Informação.

COMISSÃO EXAMINADORA

_____
Prof. Luiz Fernando Legore do Nascimento
UENP – *Campus* Luiz Meneghel

_____
Prof. Ederson Marcos Sgarbi
UENP – Campus Luiz Meneghel

_____
Prof. Fábio de Sordi Junior
UENP – Campus Luiz Meneghel

Bandeirantes, 10 de Dezembro de 2010

AGRADECIMENTOS

Agradeço a DEUS pela concepção da vida e da missão, que certamente estou desempenhando da melhor maneira.

Agradeço ao grande amor da minha vida Ana Paula, pelo seu amor, carinho, apoio e compreensão. Mulher que DEUS colocou em minha vida para que juntos possamos construir família e pregar o Evangelho.

Agradeço ao meu orientador e professor Fernando, pelo empenho e apoio na conclusão deste trabalho. Que DEUS o abençoe e sua família.

Agradeço os colegas, pela amizade e estímulo durante o cumprimento da jornada.

A todos os professores, pelo otimismo e compreensão.

Enfim, a todos aqueles que direta ou indiretamente contribuíram para que eu conseguisse realizar o melhor.

Feliz o homem que acha sabedoria, e o homem que adquire conhecimento; (de DEUS)

(Provérbios 3. 13-14)

RESUMO

Este estudo tem como objetivo mostrar o funcionamento do sistema para filtragem de conteúdo, bem como, analisar o parque tecnológico da Prefeitura Municipal de Bandeirantes, de forma a colher informações sobre a quantidade de equipamentos em rede com acesso a internet. A quantidade de banda consumida por essa rede e informações sobre o conteúdo visitado. Frente ao cenário analisado propor o uso de uma ferramenta Proxy/cache a fim de controlar os acessos e melhorar a latência da rede.

Palavras-chave: Squid. *Proxy*. Filtragem de Conteúdo.

ABSTRACT

This study aims to show the functioning of the system for content filtering, and to evaluate the technological park of the City of Bandeirantes, in order to gather information about the number of networked devices with Internet access. The amount of bandwidth consumed by this network and information about the content visited. Opposite scenario will propose the use of a tool Proxy / cache to control access and improve network latency.

Keywords: Squid. Proxy. Content Filtering.

LISTA DE FIGURAS

LISTA DE QUADROS

SUMÁRIO

1 INTRODUÇÃO

O acesso à grande rede está sendo utilizada de uma forma mais intensa, tanto como ferramenta de trabalho como também para fins de lazer, comunicação instantânea e outras finalidades diversas.

Para as empresas e instituições que utilizam a internet como sua ferramenta de trabalho, o tempo perdido com o uso desenfreado, impróprio e sem a finalidade na qual a empresa ou instituição contrata sua equipe de trabalho, acarreta muitas vezes na lentidão da rede e dispersão dos funcionários, causando, portanto, uma redução na qualidade e dificuldade na entrega dos trabalhos frente aos prazos estipulados.

Políticas e boas condutas e advertências nem sempre são a melhor opção para diminuir o mau uso da rede em um ambiente de trabalho. Assim, cabe ao administrador da rede fazer o controle dos acessos à rede *Internet*, visando à segurança da rede local, e quando necessário efetuar o bloqueio de *sites* indesejados, de constantes *downloads* que para a política adotada pela empresa são ou não permitidos.

Com o aumento dos incidentes de segurança da informação e o rápido crescimento no meio corporativo, a preocupação com segurança da informação é cada vez maior. A política de segurança é uma peça chave quando se deseja tornar um ambiente computacional mais seguro. Uma Política de Segurança é um conjunto de leis, regras e práticas que regulam como uma organização gerencia, protege e distribui suas informações e recursos (UCHÔA, 2003).

Umas das preocupações que deve ser abordada na política de segurança, diz respeito ao controle de acesso, ou seja, as diretrizes das quais o usuário pode ou não acessar e segundo seu nível de ocupação funcional na empresa ou instituição. Os mecanismos para este controle não devem ser abordados pela política de segurança, mas devem ser implementados de maneira que se faça cumprir o que nela é determinado. (NIC BR SECURITY OFFICE, 2010).

Em se tratando de acesso a internet existem várias formas de realizar o controle, a forma mais comum é o controle através de firewalls baseados em filtros de pacotes e sistemas de proxy. O proxy é um programa que fica entre a rede local e a rede pública

(internet), realizando o controle na comunicação entre os dois lados (UCHÔA; SIMEONE; SICA, 2003).

Conforme Palma e Prates (2000), cada vez mais os administradores têm que controlar e monitorar o acesso a recursos das redes de computadores. Com isto, surgiram ferramentas que implementam diversas funções, entre elas o filtro de pacotes, que trabalha na camada de rede1, e os servidores *proxy*, que trabalham na camada de aplicação2. Estas camadas baseiam-se no modelo de referência *Transfer Control Protocol/Internet Protocol* (TCP/IP) e encontram-se descritas em Péricas (2003).

Segundo Nemeth et al (2002), considerando as ferramentas de administração de redes desenvolvidas para GNU/Linux em geral, especificamente em modo *console*, pode-se dizer que somente os usuários com um conhecimento mais avançado conseguem manipulá-las e usá-las apropriadamente. Conforme Pcmaster (2005), hoje já existem interfaces mais amigáveis para o usuário poder manipular as regras e estabelecer políticas de uso dos recursos da rede. Porém especificamente para os servidores *proxy*, as ferramentas são de difícil entendimento e com uma aparência nada amigável, sendo normalmente feitas em *shell script*.

Segundo Baros (2010), o Squid é um aplicativo que está sendo melhorado continuamente, além de ser multi-plataforma, possui uma excelente estabilidade nas condições mais extremas, possuindo um imenso número de analisadores de *log*. A ferramenta Squid (proxy/cache) permite melhorar o desempenho da navegação na Internet com o *cache* que é armazenado localmente no servidor e implementa mecanismos de segurança nas alterações das suas configurações.

Em análise feita na Prefeitura Municipal de Bandeirantes, local e objeto de estudo deste trabalho foi utilizado um servidor com Sistema Operacional Linux – Kernel 2.6 com a ferramenta squid e cujos relatórios dessa ferramenta foram apresentados em um servidor de páginas web Apache, permitindo uma melhor comunicação com os administradores de rede sobre os sites visitados.

1.1 OBJETIVO GERAL

O objetivo deste trabalho é analisar o parque tecnológico da Prefeitura Municipal de Bandeirantes, de forma a colher informações sobre a quantidade de equipamentos em rede com acesso a internet. A quantidade de banda consumida por essa rede e informações sobre o conteúdo visitado. Frente ao cenário analisado propor o uso de uma ferramenta Proxy/cache a fim de controlar os acessos e melhorar a latência da rede.

1.1.2 Objetivos Específicos

Os objetivos específicos do trabalho são:

a) Análise da rede;

b) Análise do parque tecnológico;

c) Definição das políticas de utilização;

d) Implementação das políticas através de ACLs;

e) Gerar relatórios sobre os acessos ao conteúdo web.

1.3 JUSTIFICATIVA

O presente trabalho justifica-se pela necessidade de se diminuir o uso indiscriminado ao acesso a sites não condizentes com a política da Instituição, uma vez que há uma grande dificuldade no rendimento do trabalho bem como a necessidade de se ter uma rede mais estável e sem maiores custos com a compra de mais link de rede e agilidade e rapidez quanto a real necessidade do uso da rede internet.

1.4 METODOLOGIA

A pesquisa, objeto deste estudo foi de caráter exploratório e empregou as seguintes metodologias: a bibliográfica, a pesquisa de campo, buscando colocar em prática os conceitos teóricos estudados.

A pesquisa bibliográfica consistiu na consulta de livros, periódicos e outros documentos impressos como revistas especializadas. Os textos lidos e analisados tiveram como finalidade gerar conhecimento e esclarecimento sobre o tema investigado. A pesquisa bibliográfica permite ao pesquisador formular o ordenamento lógico e crítico das unidades de pensamento dos autores referenciados.

A pesquisa de campo foi na Prefeitura Municipal de Bandeirantes em seu sistema de acesso a *Internet*, utilizando ferramentas proxy/cache e analisadores de log.

1.5 ORGANIZAÇÃO DO TRABALHO

A estrutura deste trabalho está dividida em quatro capítulos, que serão explanados a seguir.

No primeiro capítulo é apresentada a introdução destacando os objetivos almejados e uma breve explicação sobre o que se pretende com este trabalho.

No segundo capítulo é apresentada a fundamentação teórica do trabalho, abordando os tópicos de gestão de redes, com alguns de seus conceitos: gerência de redes de computadores; servidor *proxy*; *cache* e seus tipos: *browse cache, proxy cache* e *transparent proxy cache*; os filtros do *proxy*; vantagens e desvantagens de um *proxy*; Squid; autenticação com os módulos compatíveis com o Squid; configuração do Squid; Apache; SARG; chpasswd; Webmin; controles de acesso e seus mecanismos e ACL. No terceiro capítulo é abordado o desenvolvimento deste trabalho, com as tecnologias utilizadas e os casos de uso. Por fim, no quarto capítulo é apresentada a conclusão do trabalho, destacando os resultados alcançados e as dificuldades encontradas.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 GESTÃO EM REDES

Segundo Lima (1997), com o aumento da presença das redes de computadores nas instituições e como conseqüência o aumento da sua importância, faz-se necessário a gerência das redes de computadores para garantir e prevenir que alguns problemas mais graves interrompam ou prejudiquem seu desempenho e funcionalidade.

2.2 GERÊNCIA DE REDES DE COMPUTADORES

Segundo Sauvé (2002), a gerência de redes de computadores é divida em cinco partes:

a) gerência de configuração – tem por objetivo analisar, monitorar mudanças referentes à infra-estrutura física e lógica e fazer a manutenção da rede. Faz a coleta de informações de configuração de equipamentos e elementos de uma rede. Gera eventos quando recursos são agregados ou eliminados da rede, permitindo manter um inventário da rede, pois faz o registro de informações de todos os elementos que possam ser gerenciados na rede;

b) gerência de faltas – é responsável pela detecção, isolamento e resolução de falhas da rede. Através da detecção de falhas é notado algum problema nos elementos, por meio de monitoração do estado de cada um. Com o isolamento de falhas, pode-se, depois de identificada a falha, verificar a causa da falha e pode-se também fazer a antecipação das falhas, ou seja, solicitar a manutenção do elemento através de alarmes, para não prejudicar o funcionamento da rede;

c) gerência de desempenho – é responsável pela monitoração de desempenho, sua análise e pelo planejamento de capacidade. A monitoração e análise de desempenho baseiam-se basicamente em indicadores, como tempo de resposta, latência da rede, disponibilidade, taxa de erros, entre outros. O planejamento de

capacidade vai basicamente demonstrar dados que sugerem a alteração no modo de operação das redes;

d) gerência de segurança – protege elementos da rede, monitorando e detectando violações da política de segurança. Preocupa-se com a proteção dos elementos da de rede, sempre com base na política de segurança pré-determinada. Faz toda a manutenção dos *logs* de segurança para detectar violações à política de segurança;

e) gerência de contabilidade – é responsável pela contabilização e verificação de limites da utilização dos elementos de rede. Monitora quais e quantos recursos da rede estão sendo utilizados, classificando por quem e quando são utilizados. E também estabelece uma escala de tarifação.

Mesmo a gestão de redes de computadores sendo de vital importância para as instituições, os administradores de rede ainda precisam de mecanismos com os quais possam monitorar e limitar as ações dos usuários das redes de computadores, principalmente no que se refere ao acesso a páginas de Internet.

2.3 PROXY

Em sua grande maioria, os navegadores de páginas *web*, fazem conexões diretas com a Internet. Mas há outra forma bem mais interessante de conexão: eles podem ser configurados para se conectarem através de um servidor *proxy*. O *proxy* é um serviço que está disponível em um ambiente servidor, que recebe requisições das estações de trabalho para conexões à Internet, onde seu papel fundamental é buscar a informação primeiramente no seu *cache* local e caso não encontre o documento requisitado, faz a busca no site solicitado pela estação de trabalho. Na segunda situação, o endereço Internet que fica registrado no servidor da página solicitada, é o do servidor *proxy*, pois o mesmo é o dispositivo que está entre a rede local e a Internet (PROXY, 2010).

Conforme Equipe Conectiva (2001), o servidor *proxy* surgiu da necessidade de ligar a rede local à grande rede de computadores, a Internet, através de um computador que provesse o compartilhamento de Internet com os demais

computadores. Pode-se fazer a seguinte analogia: rede local é uma rede interna e a Internet é uma rede externa, sendo assim, o *proxy* é o dispositivo que permite as máquinas da rede interna se conectarem ao mundo externo.

Como na maioria dos casos as máquinas da rede local não têm um endereço válido para a Internet, elas fazem a solicitação de um endereço externo para o servidor *proxy*, que encaminha a requisição à *Internet*. Caso não ache o documento solicitado em seu *cache* de *Internet*, o servidor está habilitado a fazer essa consulta, pois o mesmo tem um endereço válido na *Internet*. Sendo assim, pode-se dizer que é normal ter um servidor *proxy* diretamente ligado à Internet e com um endereço válido.

Um dos elementos mais importante de um servidor *proxy* é o seu *cache*, além é claro dos seus filtros de bloqueio ou liberação de *sites*, as *Access Control Lists* (ACLs).

O diagrama de funcionamento do Proxy pode ser observado na figura 1.

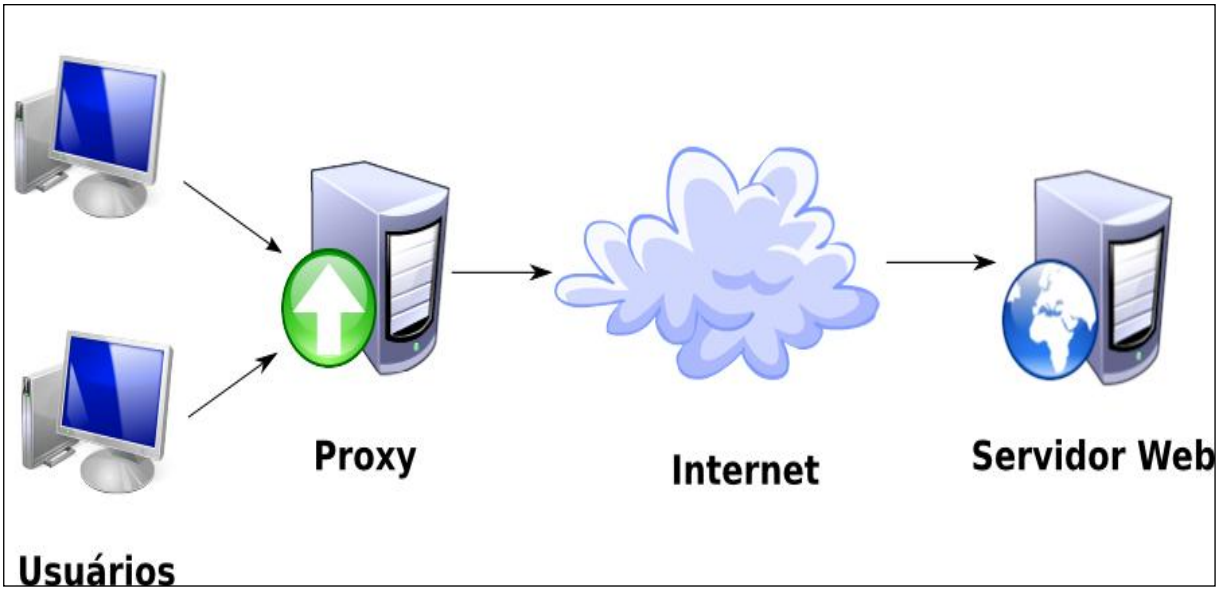

Figura 1 – Diagrama de funcionamento do Proxy – Fonte: PROXY (2010)

2.3.1 CACHE

Conforme Watanabe (2000), o *cache* é onde os arquivos requisitados pelo servidor *proxy* são armazenados e repassados posteriormente para os clientes, que

são as estações de trabalho da rede interna. Esse é um aspecto que deve ser monitorado sempre, pois pode deixar um servidor inoperante, já que são arquivos armazenados em disco e caso falte espaço em disco o servidor não vai mais funcionar. Para que isso não aconteça é necessário determinar quando os objetos serão atualizados ou removidos do *cache*, sendo que alguns desses podem permanecer sem alteração alguma por tempo indeterminado e outros podem sofrer alterações frequentemente.

Conforme Proxy (2010), visando o controle do *cache*, os servidores *proxy* utilizam algoritmos de substituição que monitoram os objetos conforme seu cabeçalho, que contém a informação de período, tamanho e histórico de acessos. Dois deles são o *Least Recent Used* (LRU), que remove objetos existentes a muito tempo e o *Least Frequently Used* (LFU), que remove os objetos menos utilizados. A utilização do espaço em disco pelo *cache* do *proxy* é controlada através desses algoritmos, juntamente com regras pré-determinadas pelo administrador.

Segundo Watanabe (2000), no caso de um objeto expirado, o servidor *web* original será consultado para revalidar o objeto. Quando o objeto tem em seu cabeçalho o campo *Last- Modified* (LM), indicando qual foi sua última alteração, o *proxy* pode usá-lo para fazer a requisição *If-Modifed-Since* (IMS) ao servidor *web* remoto, fazendo a comparação da data de alteração, identificando se o objeto foi alterado ou não e poderá atualizá-lo, caso necessário, no seu *cache*. Existem três tipos de *cache*. São eles:

a) *browse cache* – conforme Watanabe (2000), a maioria dos navegadores de *Internet* possuem um *cache* próprio, pois é muito provável que os usuários acessem os mesmos objetos frequentemente e neste caso o *cache* não é compartilhado;

b) *proxy cache* – conforme Proxy (2010), são as implementações mais utilizadas de *proxy*, e são conhecidos também como *caching web proxy*. Este disponibiliza em *cache* páginas e arquivos de servidores remotos da Internet, permitindo que os clientes da rede local acessem de forma rápida esses arquivos, considerando que a velocidade do *link* da LAN é muito maior do que o com a Internet.

Quando o *proxy cache* recebe uma solicitação de acesso a um recurso externo, como uma página da Internet, este procura primeiramente em seu *cache* local e caso

não encontre o recurso solicitado, ele imediatamente faz a requisição à Internet armazenando em seu *cache* e respondendo a solicitação do cliente. Por este motivo pode-se afirmar que o *web proxy*, além de prover segurança, provê também alto desempenho para o acesso à Internet e permite criar filtros, através de regras, dizendo o que é permitido e o que é proibido.

Segundo Watanabe (2000), a aplicação *proxy* age como um serviço intermediário entre as estações e os servidores remotos de Internet. Eles são utilizados por corporações que desejam reduzir a banda de comunicação que utilizam com a Internet;

c) *transparent proxy cache* – segundo Watanabe (2000), é utilizado especialmente por empresas provedoras de acesso à Internet, conhecidas como *Internet Service Provider* (ISP), porque permite o melhor aproveitamento de banda da Internet e não necessita fazer nenhuma configuração nas estações clientes. Conforme Proxy (2010), é uma forma de obrigarem os clientes a utilizarem o *proxy*, ou seja, além das características do *proxy cache*, ele implementa de forma transparente, por isso o nome, políticas de utilização e permite a coleta de dados estatísticos, entre outros. A transparência é implementada com a técnica de encaminhamento de portas, que é uma regra feita diretamente no *firewall* que faz o redirecionamento de todo o tráfego, por exemplo, HTTP, porta 80, para o *proxy*. Sendo assim não importa as configurações do usuário, pois sua utilização estará sempre condicionada à política de acesso pré-determinada. O *Request For Comments* (RFC) 3040, documento do *Internet Engineering Task Force* (IETF), que descreve os padrões de cada protocolo da *Internet*, define esse método como *proxy* interceptador.

2.3.2 FILTROS DE PROXY

Segundo Marcelo (2005), além do *cache*, outra característica muito importante de um servidor *proxy* são os filtros que podem ser aplicados através de regras pré-determinadas pelo administrador. Dentre elas estão às restrições a sites, configuração ou não de autenticação dos usuários e controles de acesso por horário e data. Os filtros, em geral, são conhecidos em geral como ACLs.

Conforme Watanabe (2000), os administradores podem criar os filtros dos mais simples aos mais complexos, contendo regras baseadas em diversos itens, tais como:

a) endereço de rede da estação de trabalho:

b) domínio requisitado;

c) rede de origem ou destino;

d) localização do objeto requisitado;

e) período de acesso à páginas de Internet;

f) habilitar ou não a autenticação.

Todos os filtros mencionados acima podem ser utilizados sozinhos, ou então em conjunto, mas sempre lembrando que as ACLs são analisadas de forma seqüencial. Por exemplo, se existir uma ACL com duas regras, a primeira bloqueando uma determinada página da Internet e a segunda dando permissão para todas as páginas da Internet, então a primeira regra não tem função alguma, pois a última regra invalidou a primeira.

Essas ACLs são utilizadas principalmente por corporações que queiram permitir acesso a páginas que sejam de seu real interesse, conforme as regras e a política de segurança implementada na empresa.

2.3.3 VANTAGENS E DESVANTAGENS DE UM PROXY

Segundo Watanabe (2000), algumas das principais vantagens de incentivar o uso de servidores *proxy*, são:

a) redução do tráfego de rede – são utilizadas menos requisições e respostas, sendo que o objeto do *cache* é recuperado, atualizado ou buscado do servidor uma única vez, o que reduz consideravelmente a utilização de banda por parte do cliente;

b) redução da carga dos servidores – são feitas menos requisições para os servidores *web* responderem. Por exemplo, diminui consideravelmente o congestionamento a esses servidores, quando há o lançamento de um novo produto;

c) redução de latência – possibilita a maior velocidade a resposta de requisições que são feitas ao objeto que está no *cache* do *proxy* e não diretamente ao servidor remoto;

d) possibilidade de acesso – considerando que a página de Internet solicitada está inacessível, se a página estiver como um objeto do *cache* será possível responder a requisição, apenas não possibilitando a atualização da página solicitada.

Segundo Marcelo (2005), algumas das principais desvantagens na utilização de servidores *proxy*, são:

a) poucos serviços suportados – nem todos os serviços têm suporte com os *proxies* atuais, sendo assim a relação entre o cliente e o servidor *proxy* deve ser muito bem analisada;

b) atualização de configurações em clientes – carga muito grande de modificações e/ou atualizações em clientes, principalmente em redes locais com grande número de equipamentos. Em ambientes mistos o problema pode ser maior;

c) segurança em protocolos e aplicações – o *proxy* não garante a segurança de um cliente para possíveis falhas de segurança em protocolos ou aplicações, sendo assim é necessário que o *proxy* seja implementado junto a um *firewall*.

2.4 SQUID

Conforme Marcelo (2005) e Jesus (2001), o Squid é o servidor *proxy* mais utilizado atualmente na Internet, implementando todas as características já mencionadas anteriormente.

Suporta os protocolos de comunicação HTTP, *File Transfer Protocol* (FTP) e *Gopher* e surgiu do projeto *Harvest* da ARPA. O nome Squid, que na tradução quer dizer lula foi utilizado simplesmente para distinguir um projeto do outro.

Segundo Equipe Conectiva (2001, p. 134), o Squid é um servidor *proxy* para os protocolos já mencionados anteriormente. Portanto o acesso a outros serviços como, por exemplo, o correio eletrônico, deve ser configurado com a ferramenta responsável pelo filtro de pacotes, que trabalha diretamente na camada de rede.

Conforme Baros (2006), o arquivo de configuração do Squid chamado squid.conf está organizado em *tags* que tratam de todas as configurações, tais como porta de acesso ao servidor, programa utilizado para manipulação de senhas, tamanho e estruturação do *cache*, definição e manipulação das ACLs, que vão estipular quais são os grupos de usuários a serem utilizados, quais são os arquivos com os *sites* proibidos e/ou liberados, quais são as extensões dos *downloads* proibidas, entre outras *tags.*

O que normalmente sofre maiores alterações, e com mais freqüência, são justamente as regras definidas na *tag* ACL, onde podem ser alterados os usuários, como também as listas de *sites* proibidos e/ou liberados e as extensões dos *downloads* proibidos.

O servidor Squid pode ser obtido no seu *site* oficial *Squid web proxy cache* (CHADD, et al, 2006).

2.4.1 AUTENTICAÇÃO

Conforme Marcelo (2005), a autenticação do Squid só pode ser habilitada se o mesmo for configurado em modo *proxy cache*. Caso seja configurado no modo *transparent Proxy cache*, a autenticação não é permitida com seus módulos padrões.

Segundo Vesperman (2001), os módulos de autenticação padrões do Squid são:

a) *Lightweigght Diretctory Access Protocol* (LDAP) – módulo que permite a autenticação baseada no banco de dados LDAP;

b) *Microsoft New Technology* (MSNT) – módulo que permite a autenticação baseada em um controlador de domínio Windows NT;

c) *National Center for Supercomputing Applications* (NCSA) – módulo que permite a autenticação baseado no tipo de arquivo *password* de muitos servidores *web* NCSA e segundo Marcelo (2005, p. 23) esse é o mais utilizado;

d) *Pluggable Authentication Modules* (PAM) – é um módulo de autenticação plugável e pode ser configurado para utilizar vários sistemas de autenticação;

e) *Server Message Block* (SMB) – módulo que permite a autenticação baseado em um servidor SMB tipo Microsoft NT ou Samba;

f) *New Technology Lan Manager* (NTLM) – módulo baseado em um protocolo de desafio / resposta, muito utilizado em ambientes Microsoft;

g) getpwnam – módulo baseado nos arquivos de senhas do GNU/Linux: o *passwd* e o *shadow*.

Conforme Vesperman (2001), o Squid utiliza processos auxiliares para processar as solicitações de autenticação para evitar que o mesmo seja parado ou bloqueado por causa de conexões lentas. Esses processos auxiliares são conectados por *pipes* Unix padrão e o Squid se comunica através de entradas e saídas padrão. Se o processo responder “OK”, a autenticação foi feita; se responder “ERR”, a autenticação falhou.

Como cada solicitação deve ser autenticada, o Squid guarda o nome de usuário e a senha junto com os retornos de autenticações bem sucedidas no seu *cache* por um período pré-determinado, permitindo que envie solicitações para cada página solicitando a autenticação ao usuário uma única vez.

2.4.2 CONFIGURAÇÃO

Conforme Marcelo (2005), o Squid é totalmente configurado em um arquivo chamado squid.conf, ou seja, toda e qualquer alteração nas ACLs ou alguma configuração específica deve ser feita nesse arquivo. Esse arquivo contém informações como:

a) endereço de rede do servidor *proxy* e a porta de comunicação utilizada;

b) configuração para informar o tipo de *proxy*, ou seja, *proxy cache* ou *transparent proxy*;

c) qual o tamanho utilizado pelo *cache*;

d) qual rede está liberada para acessar o servidor *proxy*;

e) tipos de ACLs;

f) entre outras;

Esse arquivo é lido de forma seqüencial quando o serviço do Squid é iniciado, portanto, as ACLs são lidas da mesma forma. Caso uma ACL se refira a um arquivo externo, esse arquivo será analisado no momento que o serviço é iniciado. Sendo assim se houver alguma alteração, independente do arquivo, o Squid deverá ser reiniciado para que as novas configurações sejam aplicadas.

Para fazer o monitoramento total do Squid são necessários alguns utilitários, dentre eles o Apache, o SARG e o chpasswd. Conforme Apache HTTP Server (2010), Apache é o servidor de páginas *web* mais utilizado no mundo, em março de 2010 o Apache era responsável pela hospedagem de 58% de todas as páginas de Internet do mundo. É compatível com sistemas GNU/Linux, Novell Netware, Microsoft, MAC OS X, entre outros sistemas operacionais.

Em Orso (2006) também é relacionado o projeto *open source* SARG. A ferramenta faz a análise dos *logs* do Squid e do *cache* do servidor *proxy*, informando ao administrador da rede onde os usuários navegaram, quanto tempo ficaram conectados, que arquivos foram baixados, qual os horários de acesso, quem foi o usuário que se autenticou, quais os *sites* proibidos que tiveram tentativas de acesso e depois gera os relatórios, que ficam disponíveis em uma página de Internet.

Também em Orso (2006) é relacionado o projeto *open source chpasswd*, que faz a alteração de senhas dos usuários do Squid com uma ferramenta para a *web,* contando com uma lista de outros colaboradores espalhados pelo mundo. Esta ferramenta foi desenvolvida em *perl script* e se comunica com um programa CGI, com o intuito de distribuir uma interface *web* através de um formulário para os usuários poderem alterar as suas senhas de acesso para o servidor *proxy*.

2.5 CONTROLE DE ACESSOS

Conforme Controle de Acesso (2010), controle de acesso em segurança, especificamente em segurança física de ambientes, é a permissão do acesso a recursos, salas, prédios, entre outros, somente a pessoas autorizadas. O controle

físico de ambientes é feito por pessoas, meios tecnológicos, cartão de acesso, abertura de porta por meio de tranca eletrônica e/ou liberado por senha, ou mecanismos de segurança como: catracas, fechaduras, chaves, entre outros.

Segundo Campos (2006), o controle às informações deve atender ao determinado nível conforme os requisitos de segurança, sempre contribuindo com o negócio da organização. O controle de acesso na segurança da informação é baseado basicamente em três processos: autenticação, autorização e contabilidade. Assim sendo, pode-se dizer que o controle de acesso é a habilidade de permitir ou negar um objeto, sendo esse uma entidade passiva, um arquivo, um sistema, entre outros, por um sujeito, uma entidade ativa, sendo esse um usuário ou processo. A autenticação identifica quem acessou o recurso, a autorização define o que o usuário pode fazer e a contabilidade informa o que esse usuário fez:

a) autenticação e identificação – parte de um processo de dois passos, categorizando quem pode acessar determinado sistema. No passo de identificação o usuário vai informar quem ele é, normalmente por um nome de usuário. No passo de autenticação ele vai informar uma credencial, por exemplo, uma senha;

b) autorização – define os direitos e permissões dos usuários. Esse processo é executado após a autenticação do usuário, determinando o que o usuário pode fazer no sistema;

c) contabilidade – coleta as informações de utilização dos usuários e dos recursos disponíveis a ele. Esse tipo de informação pode ser utilizado para gerenciamento, planejamento, entre outros. Existem dois tipos de contabilidade: em tempo real e a em *batch*. No tempo real, as informações são trafegadas no momento da utilização do recurso pelo usuário; na *batch*, as informações são gravadas e enviadas após o uso, normalmente em tempos pré-determinados. As principais informações da contabilidade são a identidade do usuário, o momento de início de utilização do recurso e o seu término.

2.5.1 MECANISMOS DE CONTROLE DE ACESSO

Os mecanismos de controle de acesso mais conhecidos são os baseados em identidade ou discricionários, os baseados em regras ou obrigatórios, e os baseados em papéis.

2.5.1.1 DAC

Conforme Silva (2004), o *Discretionary Access Control* (DAC) é uma política de controle de acesso baseada na permissão determinada pelo proprietário do recurso, por exemplo, um arquivo. O proprietário define quem tem acesso, qual a permissão e qual privilégio tem referente ao recurso. O Quadro 1, demonstra como são atribuídos os privilégios e as permissões dos sujeitos aos objetos.

	Objeto 1	Objeto 2
Sujeito 1	(read)	(read, write, execute)
Sujeito 2	-	(read, write)
Sujeito 3	(write)	-

Quadro 1 – Matriz de acesso – Fonte: Silva (2004)

Os controles discricionários podem ser utilizados empregando duas técnicas:

a) lista de controle de acesso, a ACL – responsável por definir quais são os direitos e as permissões dos usuários sobre determinado objeto ou recurso. As ACLs possibilitam um método bastante flexível para a implementação de controles discricionários;

b) controles de acesso baseados em papéis – determina as permissões e privilégios com base no papel de determinado usuário na organização. Esse método visa à simplificação do gerenciamento das permissões e privilégios dados aos usuários. As permissões de acesso e direitos sobre determinados objetos são dados para qualquer grupo ou indivíduo. Um indivíduo pode pertencer a um ou mais grupos e podem adquirir permissões cumulativas ou serem eliminadas algumas permissões, dos grupos que ele não pertence.

2.5.1.2 MAC

Segundo Silva (2004), o *Mandatory Access Control* (MAC) implementa uma política obrigatória, ou seja, as regras de controle de acesso são impostas por uma autoridade central, normalmente o administrador do sistema, que especifica regras de controle de acesso para recursos e informações, garantindo que as mesmas sejam incontornáveis. Sendo assim esse mecanismo é bem mais complexo para implementar, pois utiliza política multinível e devido a sua rigidez com as regras de controle e também com relação as limitações dos seus modelos.

As políticas multinível são baseadas na classificação que estão submetidos os sujeitos e os objetos. Uma forma de viabilizar a implementação da política multinível é a sugestão de construir reticulados com rótulos de segurança, sendo que os rótulos de segurança contêm níveis de sensibilidade e categoria. As categorias são os compartimentos específicos do sistema que pertencem às informações de uma determinada organização. Os níveis de sensibilidade atribuídos às informações são derivadas diretamente da classificação utilizada.

Os rótulos de segurança (produto vetorial do rótulo de segurança é: rótulo de segurança = nível de segurança X categoria), são o produto vetorial do conjunto de níveis de sensibilidade pelo conjunto de categorias, sendo que a categoria é o conjunto de todos os subconjuntos formados a partir das categorias pré-definidas no modelo.

2.5.1.3 RBAC

Segundo Silva (2004), os modelos baseados em papéis, *Role-Based Access Control* (RBAC), intermedeiam o acesso a informação baseado nas atividades que os usuários desempenham no sistema, podendo o usuário desempenhar papéis diferentes no sistema. Um papel pode ser definido como um conjunto de atividades e

responsabilidades atribuídas a um cargo ou função dentro de uma organização. Sendo assim, os usuários têm autorização para exercer papéis, e os papéis recebem as permissões.

A Figura 2 demonstra o modelo básico do RBAC.

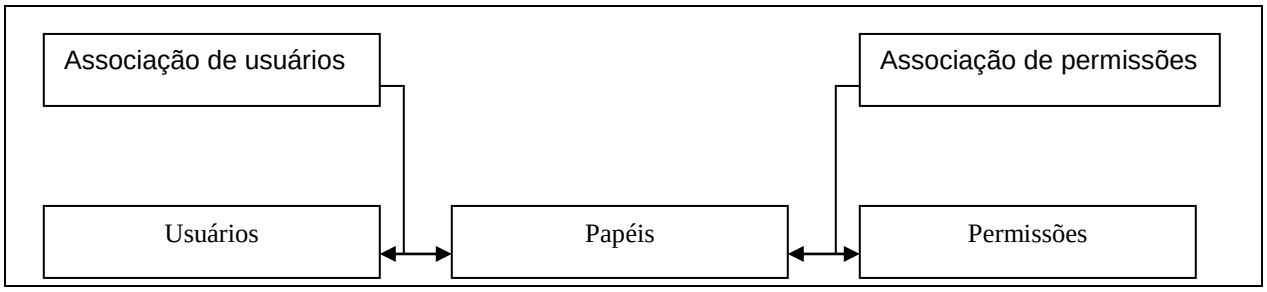

Figura 2 – Modelo básico de RBAC. Fonte: Silva (2004)

O RBAC tem por objetivo facilitar a gerência de autorização, isso por que quando o usuário tem alguma mudança nas suas atribuições, sendo eliminado de um papel e atribuído a outro, a manutenção das permissões dos papéis não sofre mudanças. Normalmente o RBAC implementa o princípio de privilégio mínimo, ou seja, um usuário ativa somente o subconjunto de papéis que precisa para acessar determinado recurso ou informação e essa ativação pode ou não ter restrições.

2.5.2 ACL

Conforme Marcelo (2005), o *web proxy* permite ou não a autenticação, o que vai possibilitar a implementação de perfis de acesso à Internet, com o bloqueio e/ou liberação de serviços. Já o *proxy* transparente, não permite que seja configurada a autenticação, somente com algum módulo ou sistema a mais que possibilite a autenticação.

Para poder implementar esse tipo de controle por usuários é necessário a implementação de políticas de acesso, que são as populares ACLs.

Segundo Silva (2004), se for considerado uma coluna do Quadro 1, veremos que a relação de todos os sujeitos com seus respectivos direitos de acesso sobre um determinado objeto, correspondem a coluna, formando uma lista de controle de acesso, ou uma ACL, do objeto considerado. As ACLs são uma forma de representação da matriz de acesso. O Quadro 2 apresenta um conjunto de ACLs onde cada lista corresponde ao controle do objeto correspondente.

Objetos	**Listas de controle de acessos**
Objeto 1	sujeito 1 (read), sujeito 2 (write, read) sujeito 3 (read, write, execute)
Objeto 2	sujeito 2 (read, execute), sujeito 4 (write)
Objeto 3	Sujeito 3 (read, write, execute), sujeito 1 (execute)

Quadro 2 – Conjunto de ACLs . Fonte: Silva (2004)

A ACL de um sujeito permite uma fácil revisão dos acessos autorizados dele a um determinado objeto ou recurso. Outra operação que pode facilmente ser implementada com uma ACL é a revogação de todos os direitos de acesso de um usuário sobre um objeto, para isto basta substituir a ACL atual por uma lista vazia.

Sendo assim, para determinar os acessos aos qual o sujeito está autorizado, todas as listas de controles do sistema devem ser percorridas, para fazer a revisão do acesso. A revogação de todos os acessos também requer que todas as listas de controle sejam analisadas e, eventualmente, alteradas.

3. DESENVOLVIMENTO DO TRABALHO

Neste capítulo são apresentadas técnicas e ferramentas utilizadas para permitir o gerenciamento da rede baseadas principalmente na regras de ACLs do servidor Squid.

3.1 REQUISITOS PRINCIPAIS DO PROBLEMA TRABALHADO

A aplicação permitiu configurar o servidor *proxy* Squid. Pode-se citar como requisitos principais da aplicação, estipulando-os em Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF):

a) permitir a alteração do arquivo de configuração do servidor *proxy* Squid através de formulários e múltiplas seleções utilizando *checkboxes* e *radio buttons* (RF);

b) permitir que sejam cadastrados usuários nos grupos: VIP, moderado e restrito (RF);

c) permitir que as senhas dos usuários da ferramenta sejam armazenadas no sistema pelo algoritmo MD5, que faz um *hash* da senha e é feito pelo utilitário MD5 do GNU/Linux (RF);

d) permitir que o servidor *proxy* seja reiniciado para aplicar as novas configurações (RF);

e) permitir a alteração de usuários de um grupo para outro (RF);

f) permitir a exclusão de usuários dos grupos (RF);

g) permitir o cadastro de páginas, extensões de arquivos e palavras proibidas para os grupos moderado e restrito (RF);

h) permitir o bloqueio de computadores pelos seus endereços de rede (RF);

i) permitir a liberação de páginas para o grupo restrito (RF);

j) permitir a liberação e bloqueio de portas de comunicação (RF);

k) permitir recriar o *cache* do servidor *proxy* (RF);

l) permitir a monitoração em tempo real do *log* do Squid (RF);

m) permitir a configuração dos parâmetros da aplicação estipulando onde os arquivos de configuração se encontram no servidor (RF);

n) permitir a geração e visualização de relatórios de acesso com a utilização do SARG (RNF);

3.2 ESTUDO DE CASO – PREFEITURA MUNICIPAL DE BANDEIRANTES

O servidor da prefeitura, são utilizados diariamente por 51 usuários e através de seus Desktops ou Notebooks, acessam a internet todos os dias, abaixo descritos por setor:

SETOR	Quantidade de Computadores	
	Desktops	Notebooks
Processamento de Dados	01	
Receita	04	
Tesouraria	03	
Recursos Humanos	05	
Jurídico	02	01
Chefe de Gabinete	01	
Desenvolvimento Urbano e Habitação	01	
Departamento de Compras	02	01
Administração	02	01
Contabilidade	06	
Licitação	04	
Indústria Comércio e Turismo	01	
Protocolo	01	01
Secretaria	02	
Agricultura	04	02
Obras	05	
PROCON	02	

Quadro 3 - Usuários do servidor da Prefeitura. Fonte: da pesquisa (2010)

3.2.1 CASO DE USO DE ACESSO USUÁRIO SQUID

Neste capítulo será descrito o caso de uso do servidor *proxy*, configurado pela ferramenta *web* utilizada para administrar o *proxy*, conforme descrito na Quadro 3, no caso de uso do acesso do usuário ao Squid.

No Quadro 4, é apresentado o caso de uso do acesso *web* via *proxy.*

Descrição	Usuário acessa a Internet da rede interna, via *proxy*, configurado pela ferramenta para administrar o *proxy.*
Autor	Usuário
Pré-condição	Administrador deve configurar o servidor Squid pela ferramenta *web* para administrar o *proxy*.
Fluxo principal	a) verificar junto ao administrador que tipo de *proxy* foi configurado; - *proxy* autenticado; - *proxy* transparente; b) abrir o navegador de Internet; - caso o *proxy* configurado seja o autenticado, usuário deverá inserir seu nome de usuário e a senha, após seguir para passo “c”; - caso o *proxy* configurado seja o transparente, seguir para o passo “c”; c) navegar na Internet.
Fluxo alternativo (a)	a) *proxy* autenticado: - abrir as configurações do navegador de Internet e configurar o endereço e a porta de comunicação do servidor proxy; - seguir para o passo “b” do fluxo principal. b) *proxy* transparente: - seguir para o passo “b” do fluxo principal.
Pós-condição	Usuário terá acesso à internet

Quadro 4 – Caso de uso do acesso *web* via *Proxy* - Fonte: da pesquisa (2010)

3.2.2 CASO DE USO DE ACESSO ADMINISTRADOR

Neste capítulo serão descritos os casos de uso da ferramenta *web* para administrar o *proxy*, conforme descrito na Quadro 4, caso de uso do acesso do administrador.

No Quadro 5, é apresentado o caso de uso acessar a aplicação.

Descrição	Administrador acessa o endereço da aplicação no navegador *web*.
Autor	Administrador.
Pré-condição	Configuração do ambiente operacional e administrador deverão possuir nome de usuário e senha para acessar a ferramenta.
Fluxo principal	a) abrir o navegador Internet; b) abrir o endereço da aplicação; c) digitar o nome de usuário e senha; - usuário ou senha inválido; d) clicar no botão entrar; e) acesso a aplicação.
Fluxo exceção (c)	a) usuário ou senha inválido: - se aparecer à mensagem "Usuário e/ou senha inválido(s)"; - fazer o passo “c” e os seguintes do fluxo principal.
Pós-condição	Administrador terá acesso à aplicação.

Quadro 5 – Caso de uso acessar a aplicação - Fonte: da pesquisa (2010)

3.2.3 CONFIGURAÇÃO DO SQUID

No Quadro 6, é apresentado o caso de uso criar configuração do Squid e reiniciar.

Descrição	Permitir configurar o Squid, criar o *cache* e reiniciar o mesmo para que as alterações tenham efeito.
Autor	Administrador.
Pré-condição	Criar permissões para linha de comando e configurar os parâmetros, conforme ambiente operacional
Fluxo principal	a) clicar no link criar; b) no campo IP/Porta, inserir o endereço da rede interna do servidor *proxy* e a porta de comunicação utilizada; c) selecionar o tipo de *proxy*: - *proxy* autenticado; - *proxy* transparente; d) no campo tamanho do *cache* em *bytes*, inserir o tamanho do diretório de *cache* do servidor *proxy*; e) no campo rede liberada / máscara, inserir o endereço de rede interna, com sua respectiva máscara, que será liberada para acessar a Internet; f) no campo bloqueio de *downloads*, marcar a seleção se deseja configurar o bloqueio de *downloads* da Internet; g) no campo bloqueio de *sites*, marcar a seleção se deseja configurar o bloqueio de *sites* à Internet; h) no campo bloqueio de palavras chaves, marcar a seleção se deseja configurar o bloqueio de palavras chaves à Internet; i) no campo bloqueio de computadores, marcar a seleção se deseja configurar o bloqueio de computadores à Internet da rede interna; j) clicar no botão gerar script; k) clicar no link comandos; - clicar no botão criar cache; l) clicar no botão reconfigura o Squid.
Fluxo alternativo (c)	a) *proxy* autenticado: - será configurado o Squid com o módulo de autenticação; - será utilizado o módulo de autenticação NCSA. b) *proxy* transparente: - será configurado o Squid para ser um *proxy* transparente; - deverão ser configuradas regras para o *proxy* transparente no *firewall*.
Fluxo alternativo (k)	a) clicar no *link* comandos; b) clicar no botão gerar, da opção criar *cache*; c) caso seja a primeira configuração, o *cache* do Squid deverá ser criado.
Pós-condição	Squid configurado.

Quadro 6 – Caso de uso configuração do Squid - Fonte: da pesquisa (2010)

3.2.4 CADASTRO DE USUÁRIOS NOS GRUPOS

No Quadro 7, é apresentado o caso de uso cadastrar usuários nos grupos.

Descrição	Permitir o cadastro de usuários nos devidos grupos de acesso. Permitir a consulta de usuários nos grupos de acesso e a alteração de grupos de acesso.
Autor	Administrador.
Pré-condição	Squid configurado.
Fluxo principal	a) clicar no *link* usuários; b) clicar no *link*: - cadastro de usuários; consulta / alteração de usuários;
Fluxo alternativo (b)	a) cadastro de usuários: - no campo nome / senha, inserir o nome de usuário e sua senha de acesso à Internet; - selecionar o grupo de usuários a que pertence; - clicar no botão cria usuário; - será apresentada uma mensagem: “Usuário: nome do usuário, inserido no grupo de usuários selecionado com sucesso”. b) consulta / alteração de usuários: - consultar o usuário, basta acessar esse *link*; - alterar o usuário de grupo: - selecionar o usuário a ser apagado do grupo; - apagar o nome de usuário selecionado, com a tecla *delete*; - usuário será apagado do grupo; - clique no botão voltar; - clicar no botão atualizar nome do grupo escolhido; - inserir o mesmo nome de usuário que foi excluído anteriormente, no grupo escolhido; - clicar no botão atualizar nome do grupo escolhido; - clique no botão voltar; - usuário será mostrado no grupo que foi inserido.
Fluxo exceção (a)	a) Se for apresentada mensagem de erro: - “Por favor, preencha o campo usuário”; - “Por favor, preencha o campo senha”; - “Por favor, selecione o grupo desejado”; b) fazer o passo “b” do fluxo principal, e os seguintes até não apresentar erro.
Pós-condição	Usuários inseridos em seus grupos de acesso conforme política de acesso à Internet.

Quadro 7 – Caso de uso cadastrar usuários nos grupos - Fonte: da pesquisa (2010)

3.2.5 BLOQUEAR DOWNLOADS POR EXTENSÕES

No Quadro 8 é apresentado o caso de uso bloquear *downloads* por extensões.

Descrição	Permitir o bloqueio de *downloads* filtrados pelas extensões dos arquivos, tanto do grupo de usuários moderados como do grupo de usuários restritos.
Autor	Administrador.
Pré-condição	Squid configurado para bloquear *downloads.*
Fluxo principal	a) clicar no *link downloads*; b) inserir as extensões de arquivos que devem ser bloqueadas na área de texto, uma abaixo da outra; clicar no botão salvar.
Pós-condição	Squid configurado para bloquear *downloads* conforme lista de extensões definida.

Quadro 8 – Caso de uso bloquear *downloads* por extensões - Fonte: da pesquisa (2010)

3.2.6 Bloquear Sites, Palavras e Máquinas

No Quadro 9 é apresentado o caso de uso bloquear *sites*, palavras e máquinas.

Descrição	Permitir o bloqueio de palavras e *sites* para o grupo de usuários moderado e também permitir o bloqueio de máquinas pelo seu endereço de rede.
Autor	Administrador.
Pré-condição	Squid configurado para bloquear palavras, sites e máquinas.
Fluxo principal	a) palavras; b) *sites*; c) máquinas.
Fluxo alternativo (a)	a) clicar no *link* palavras; b) inserir as palavras que devem ser bloqueadas na área de texto, uma abaixo da outra; c) clicar no botão salvar
Fluxo Alternativo (b)	a) clicar no *link sites*; b) inserir os *sites* ou parte do endereço que devem ser bloqueados na área de texto, um abaixo da outro; c) clicar no botão salvar.
Fluxo Alternativo (c)	a) clicar no *link* máquinas; b) inserir os endereços de rede que devem ser bloqueados na área de texto, um abaixo da outro; c) clicar no botão salvar.
Pós-condição	Squid configurado para bloquear palavras proibidas, *sites* proibidos e máquinas não permitidas ao acesso à Internet, conforme lista inserida.

Quadro 9 – Caso de uso bloquear palavras, *sites* e máquinas - Fonte: da pesquisa (2010)

3.2.7 BLOQUEAR/LIBERAR PORTAS DE COMUNICAÇÃO

No Quadro 10 é apresentado o caso de uso bloquear/liberar portas de comunicação.

Descrição	Fazer a liberação ou bloqueio de portas de comunicação que são utilizadas em aplicações no navegador de Internet e fazem as requisições em cima da porta de comunicação padrão da Internet.
Autor	Administrador.
Pré-condição	Squid configurado para bloquear/liberar portas de comunicação utilizadas diretamente no navegador de Internet.
Fluxo principal	a) clicar no *link* portas; b) liberar portas; c) bloquear portas.
Fluxo alternativo (b)	a) inserir no campo liberar porta, o número da porta de comunicação a ser liberada pelo *proxy*; b) clicar no botão liberar; consultar as portas de comunicação liberadas na área de texto das portas liberadas.
Fluxo Alternativo (c)	a) inserir no campo bloquear porta, o número da porta de comunicação a ser bloqueada pelo *proxy*; b) clicar no botão bloquear; c) consultar as portas de comunicação liberadas na área de texto das portas bloqueadas.
Pós-condição	Squid configurado para bloquear/liberar portas de comunicação utilizadas diretamente no navegador de Internet, conforme lista inserida de portas bloqueadas e liberadas.

Quadro 10 – Caso de uso bloquear/liberar portas de comunicação - Fonte: da pesquisa (2010)

3.2.8 LIBERAR SITES PARA GRUPO RESTRITO

No Quadro 11 é apresentado o caso de uso liberar *sites* para grupo restrito.

Descrição	Permitir a liberação de *sites* na *web* para usuários do grupo restrito.
Autor	Administrador.
Pré-condição	Squid configurado.
Fluxo principal	a) clicar no *link* domínios; b) inserir parte ou o endereço completo dos *sites* que devem ser bloqueados na área de texto, um abaixo do outro; c) clicar no botão salvar.
Pós-condição	Lista de *sites* liberados para o grupo de usuários restritos.

Quadro 11 – Caso de uso liberar *sites* para grupo restrito – Fonte: da pesquisa (2010)

3.2.9 CONFIGURAR/GERAR RELATÓRIOS DE ACESSO

No Quadro 12 é apresentado o caso de uso configurar/gerar relatórios de acesso.

Descrição	Configuração e geração dos relatórios de acesso à Internet.
Autor	Administrador.
Pré-condição	Configurar parâmetros, gerar permissões para linha de comando e o administrador deve estar conectado a ferramenta.
Fluxo principal	a) clicar no link relatórios; b) inserir no campo caminho relatórios o caminho absoluto de onde os relatórios serão gerados; c) inserir no campo título do relatório o título do relatório para quando o mesmo for gerado; d) clicar no botão Alterar; e) clicar no botão gravar sarg.conf: - gerar relatórios; - consultar relatórios.
Pós-condição	Configurado SARG, para gerar os relatórios de acesso à Internet pelo *proxy*. Relatórios prontos para serem gerados.

Quadro 12 – Caso de uso configurar/gerar relatórios de acesso. Fonte: da pesquisa (2010)

No caso do servidor da Prefeitura Municipal de Bandeirantes, ao inserir uma ACLs no Squid. conf., o Squid as lê de cima para baixo e quando encontra alguma que se aplique ele para. Foram criadas 3 (três) ACLs: Acesso Total; Acesso Restrito e Bloqueado. Estes arquivos apresentam os seguintes conteúdos:

Acesso Total: IPs dos clientes que terão acesso total à internet. Não passarão por nenhuma restrição no Squid.

Acesso Restrito: IPs dos clientes que passarão pelo bloqueio de sites estabelecido na ACL seguinte.

Bloqueado: Lista de palavras que o Squid bloqueará se forem encontradas na URL.

Estas três ACLs são declaradas no Squid. conf. desta maneira:

Acesso Total: acl acesso_total src "/etc/squid/acesso_total".

Acesso Restrito: acl acesso_restrito src "/etc/squid/acesso_restrito".

Bloqueado: acl bloqueado url_regex -i "/etc/squid/bloqueado".

Com as regras declaradas, vamos ativá-las dessa maneira:

- http_access allow acesso_total
- http_access deny bloqueado
- http_access allow acesso_restrito
- http_access deny all

O Squid as lê desta forma: a primeira regra que o Squid lê (http_access allow acesso_total) diz que será LIBERADO acesso a quem estiver com o IP cadastrado no arquivo "/etc/squid/acesso_total". Então, quem ele encontra aqui já é liberado e não passa mais pelas outras ACLs seguintes. Por isso o acesso é direto e total.

A segunda regra que ele encontra (http_access deny bloqueado) diz que será NEGADO o acesso às URLs que coincidirem com as palavras que estão no arquivo "/etc/squid/bloqueado". Se, por exemplo, neste arquivo tiver a palavra sexo, qualquer site que tenha esta palavra na sua URL não será acessado, como em www.uol.com.br/sexo, www.sexomais.com.br, etc. Mas atenção neste detalhe. O Squid vem lendo o arquivo de cima para baixo e só chegará à segunda regra quem

não cair na primeira, ou seja, quem não tiver o IP cadastrado no arquivo de acesso total.

A terceira regra que o Squid lê (http_access allow acesso_restrito) diz que será LIBERADO acesso a quem tiver com o IP cadastrado no arquivo "/etc/squid/acesso_restrito". Como na terceira regra só chega quem não caiu na regra anterior, o acesso pode ser liberado tranquilamente.

A quarta e última regra (http_access deny all) nega o acesso a qualquer IP de qualquer máscara (0.0.0.0/0.0.0.0), pois ela já vem declarada no início das ACLs (acl all src 0.0.0.0/0.0.0.0).

As regras implantadas na Prefeitura Municipal de Bandeirantes são:

- Navegação Livre:

```
##-Range de IPs - Navegação Livre-############################
    acl ip_nlivre src "/usr/local/squid/etc/ip_nlivre.txt"
                http_access allow ip_nlivre all
```

- Palavras Chaves Liberadas:

```
### Palavras Chaves Liberadas-#################################
 acl txtlivres url_regex "/usr/local/squid/etc/txtlivres.txt"
               http_access allow txtlivres all
```

- Sites Liberados:

```
### Sites Liberados-###########################################
acl siteslivres url_regex "/usr/local/squid/etc/siteslivres.txt"
              http_access allow siteslivres all
```

- Palavras Chaves Bloqueadas:

```
### Palavras Chaves Bloqueadas -###############################
   acl txtbloq url_regex "/usr/local/squid/etc/txtbloq.txt"
                http_access deny txtbloq all
```

- Bloqueio por Sites:

```
### Bloqueio por Sites (URLs) -################################
  acl sitesbloq url_regex "/usr/local/squid/etc/sitesbloq.txt"
               http_access deny sitesbloq all
```

- Bloqueio por Downloads:

```
  ### Bloqueio por downloads -#################################
acl downloadsbloq url_regex "/usr/local/squid/etc/downloadsbloq.txt"
              http_access deny downloadsbloq all
```

-Range de IPs – Navegação Restrita:

```
##-Range de IPs - Navegação Restrita-################################
    acl ip_nfiltrada src "/usr/local/squid/etc/ip_nfiltrada.txt"
                http_access allow ip_nfiltrada all
```

- URLS – Conteúdo Restrito:

```
##-URLS - Conteúdo Restrito-#########################################
acl urlsrestritas url_regex "/usr/local/squid/etc/urlsrestritas.txt"
                http_access allow urlsrestritas all
```

- Range de IPs – Navegação Restrita:

```
##-Range de IPs - Navegação Restrita-################################
                  acl ip_nrestrita src 10.3.0.0/24
                    http_access deny ip_nrestrita
```

3.3 ANÁLISE DE TRÁFEGO DA REDE

Antes da implantação das ferramentas de tráfego funcionava com a seguinte capacidade, figura 3, onde se pode observar as taxas de entrada *(incoming rates*) e as taxas de saída *(outcoming rates).* A taxa de saída (*outcoming rates*) da placa do servidor interna estava operando com 2062 kbit por segundo durante os horários de pico além da redução do broadcast e consequentemente a quantidade de pacotes trafegados.

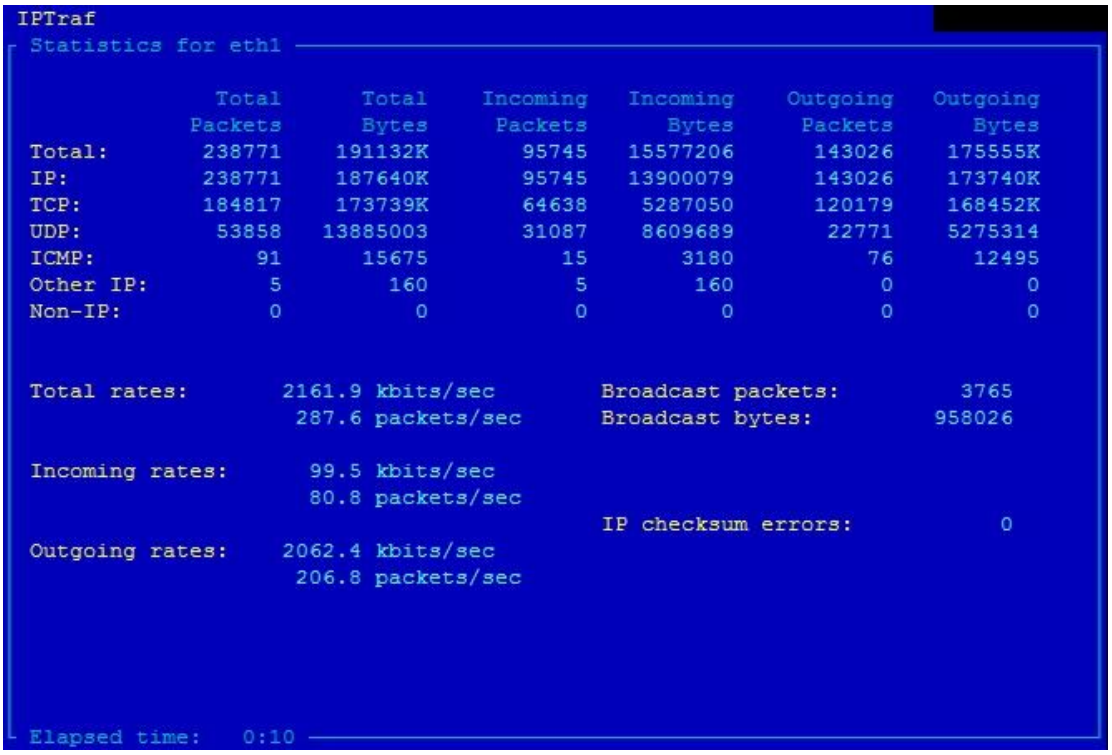

Figura 3. Antes da implantação da ferramenta de filtragem. Fonte: pesquisador (2010)

Após a implantação e mudanças das ferramentas, o fluxo de dados na rede diminuiu consideravelmente, devido ao bloqueio de diversos sites irrelevantes ao serviço prestado na Prefeitura, a partir dai o tráfego de rede passou a ser operado da seguinte forma:

.

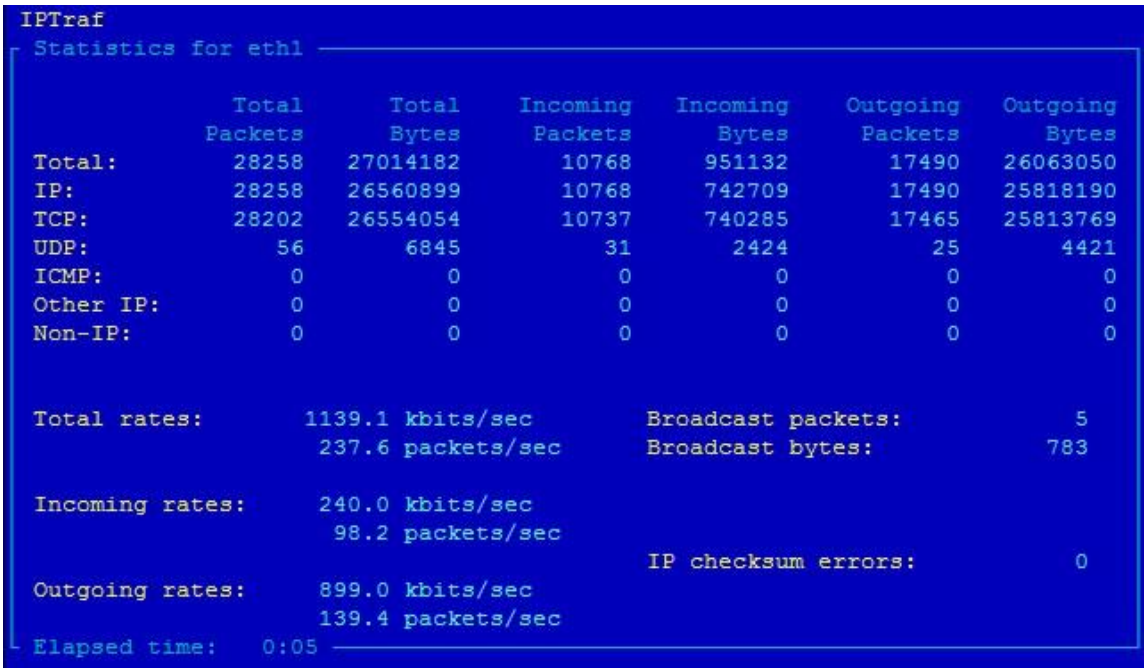

Figura 4. Depois da implantação da ferramenta de filtragem. Fonte: Pesquisador (2010)

Antes o Total rates de entrada e saída era de 2161.9 Kbits/sec e passou a ser de 1139.1 Kbits/sec, mesmo ocorrendo em horário de pico, houve uma redução do broadcast que era de 3765 e passou a ser 5, diminuindo a quantidade de pacotes trafegados.

3.4 RELATÓRIO PARCIAL DA FERRAMENTA SARG

O relatório da ferramenta SARG, permite ser visto os acessos, o endereço IP da máquina que gerou a requisição e a negação de serviço para uma url específica, dados estes colhidos após a implantação das ferramentas de controle.

Pode-se analisar pelas figuras 3 e 4, que o fluxo de dados na rede diminuiu consideravelmente após a implantação das ferramentas de filtragem, onde utilização o registro de acesso, conforme figura 5 (abaixo), como base de dados .

Squid Analysis Report Generator

Registro de acessos - Prefeitura Municipal de Bandeirantes

Periodo: 2010May14-2010May18

Usuario: 10.3.0.60

Ordem: BYTES, reverse

Usuario Relatorio

LOCAL ACESSADO	CONEXÃO	BYTES	%BYTES	IN-CACHE-OUT		TEMPO GASTO	MILISEG	%TEMPO	
bn.uol.com.br	2.41K	9.46M	7.72%	0.08%	99.92%	00:12:02	722.14K	3.44%	
portal.rpc.com.br	2.43K	8.40M	6.85%	65.37%	34.63%	00:11:54	714.88K	3.40%	
noticias.uol.com.br	2.56K	6.31M	5.15%	1.40%	98.60%	00:41:31	2.49M	11.86%	
contenti1.espn.com.br	125	4.08M	3.33%	22.35%	77.65%	00:01:33	93.02K	0.44%	
n.i.uol.com.br	11.88K	3.95M	3.22%	94.87%	5.13%	00:04:40	280.01K	1.33%	
www.midiasemmedia.com.br	50	3.93M	3.21%	42.08%	57.92%	00:02:08	128.05K	0.61%	
adclient-uol.lp.uol.com.br	1.13K	3.87M	3.15%	98.27%	1.73%	00:01:10	70.29K	0.33%	NEGADO
blog.miltonneves.ig.com.br	30	3.29M	2.68%	0.00%	100.00%	00:02:00	120.10K	0.57%	
www.diariosp.com.br	1.77K	3.03M	2.48%	19.64%	80.36%	00:34:47	2.08M	9.93%	
f.i.uol.com.br	115	2.79M	2.27%	0.44%	99.56%	00:02:04	124.08K	0.59%	
www.lancenet.com.br	773	2.32M	1.89%	8.61%	91.39%	00:02:52	172.52K	0.82%	
www.gazetadopovo.com.br	1.20K	2.32M	1.89%	65.65%	34.35%	00:05:48	348.66K	1.66%	
ads.rpc.com.br	65	2.25M	1.84%	18.92%	81.08%	00:01:30	90.13K	0.43%	
www.gazetaesportiva.com.br	941	2.16M	1.76%	52.56%	47.44%	00:02:51	171.11K	0.81%	
zerohora.clicrbs.com.br	629	2.13M	1.74%	6.69%	93.31%	00:10:13	613.14K	2.92%	
www.clicrbs.com.br	1.44K	1.99M	1.62%	15.21%	84.79%	00:16:26	986.70K	4.70%	
www.tce.pr.gov.br	80	1.91M	1.56%	49.53%	50.47%	00:00:35	35.92K	0.17%	

Figura 5 – Relatório de Registro de Acesso – Fonte: Pesquisador (2010)

É gerado um relatório para que o administrador, possa acompanhar quais os sites visitados, a lista, dos tops sites mais visitados, tempo de conexão, com entradas e saídas de dados e se houve ou não bloqueio em algum site solicitado pelo cliente com tempo de conexão, deixando a rede com um trafego otimizado.

A ferramenta SARG pode ainda ser melhorada a partir da identificação do usuário. Assim, ao invés de gerar relatórios por IP da máquina, podem-se gerar relatórios por usuários, desde que cada um possua um login e senha para ter acesso a rede.

Toda manutenção é feita on line, onde o servidor utiliza um ip fixo e serviço de ssh para acesso via modo texto.

O **SARG** (sigla para *Squid Analysis Report Gene*rator) é um gerador de relatórios que provê informações sobre a atividade dos usuários do squid com riqueza de detalhes e interface agradável.

Uma vantagem importante observada nessa ferramenta *web* em comparação com seu correlato chamado protótipo de ferramenta *web* para gerenciamento de *firewall* (BORSCHEID, 2005), é que a ferramenta *web* implementada aqui não é baseada em um filtro de pacotes, mas sim em um filtro de conteúdo.

Com relação ao seu correlato chamado Webmin (ZAGO, 2007), a ferramenta *web* desenvolvida, oferece uma linguagem mais acessível e menos técnica, além de ser uma ferramenta bem mais específica que o correlato.

O correlato chamado SARG (ORSO, 2006), oferece somente a funcionalidade de gerar os relatórios de acesso do *proxy*, sendo assim, foi utilizado para essa função.

4 CONSIDERAÇÕES FINAIS

A implantação deste modelo permitiu à Prefeitura Municipal de Bandeirantes terem um controle em relação às ações de suas estações de trabalho ligadas à internet. Os acessos a sites não apropriados foram bloqueados com as devidas regras.

Com a implementação do presente modelo e com base na análise dos resultados obtidos, verificou-se que é possível a aplicação de técnicas de análise a fim de gerar regras de bloqueios para o *proxy Squid* com o auxílio das metodologias estatísticas as quais levam em conta o perfil de acesso de cada usuário e também com o uso das expressões regulares como forma de operação sobre cadeia de caracteres com o intuito de identificar padrões de palavras.

Ainda é necessário, melhorar a ferramenta, pois o banco de palavras e sites precisa ser sempre renovado, e isso pode ser feito através de solicitações feitas pelos usuários à administração bem como a leitura dos relatórios dos sites acessados através da ferramenta SARG.

A implementação de um servidor Proxy/cachê permitiu maior agilidade da rede, diminuição de tráfego e comprometimento da equipe de trabalho com o trabalho em si. Deixando, portanto de usar para fins inadequados a filosofia da empresa.

Alguns sites não podem passar por Proxy para trabalhar, a exemplo disso tem-se o sistema da Caixa Econômica Federal, Conectividade Social. O que necessita de regras feitas no firewall para impedir que o acesso a esse conteúdo através do Proxy.

Para a Prefeitura Municipal de Bandeirantes, objeto de estudo, apresento como trabalhos futuros a criação de servidores proxy/cache em outros pontos de acesso espalhados pela cidade em forma de secretarias e postos de atendimento.

REFERÊNCIAS

APACHE HTTP SERVER. In: WIKIPEDIA, a enciclopédia livre. [S.l.]: Wikimedia Foundation, 2010. Disponível em: < http://en.wikipedia.org/wiki/Apache_server>. Acesso em: 22 out. 2010.

BAROS, E. B. **Configurando um Squid "ninja"**. [S.l.], [2010]. Disponível em: <http://www.linuxman.pro.br/squid/>. Acesso em: 20 out. 2010.

BORSCHEID, R. M. **Protótipo de aplicação web para gerenciamento de firewall em Linux**. 2005. 52 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

CAMPOS, A. L. N. **Sistema de segurança da informação:** controlando os riscos. Florianópolis: Visual Books, 2006.

CHADD, A. et al. **Squid web proxy cache**. [S.l.], [2006]. Disponível em: <http://www.squid-cache.org>. Acesso em: 21 out. 2010.

CONTROLE DE ACESSO. In: WIKIPEDIA, a enciclopédia livre. [S.l.]: Wikimedia Foundation, 2010. Disponível em: <http://pt.wikipedia.org/wiki/Controle_de_acesso>. Acesso em: 24 out. 2010.

EQUIPE CONECTIVA. **Segurança de redes:** firewall. [Curitiba]: Conectiva S.A., 2001.

JESUS, D. C. S. de et al. **Implantando WCCP na hierarquia de proxies da RNP**. [Rio de Janeiro], [2001]. Disponível em: < http://www.rnp.br/newsgen/0103/wccp.html>. Acesso em: 20 out. 2010.

LIMA, M. M. de A. E. **Introdução a gerenciamento de redes TCP/IP**. [Rio de Janeiro], [1997]. Disponível em: <http://www.rnp.br/newsgen/9708/n3-2.html>. Acesso em: 21 out. 2010.

MARCELO, A. **Squid:** configurando o proxy para Linux. 4. ed. Rio de Janeiro: Brasport, 2005.

NIC BR SECURITY OFFICE. **Práticas de Segurança para Administradores de Redes Internet, Versão 1.2**. [S.l.]. Dispon´ıvel em: <http://www.nbso.nic.br-/docs/seg-adm-redes/seg-adm-redes.pdf>. Acesso em: 20 out. 2010

NEMETH, E. et al. **Manual do administrador do sistema Unix**. 3. ed. Tradução Edson Furmankiewicz. Porto Alegre: Bookman, 2002.

ORSO, P. **SARG:** Squid Analysis Report Generator. [S.l.], [2006]. Disponível em: <http://sarg.sourceforge.net>. Acesso em: 22 out. 2010.

PÉRICAS, F. A. **Redes de computadores:** conceitos e a arquitetura Internet. Blumenau: Edifurb, 2003.

PROXY. In: WIKIPEDIA, a enciclopédia livre. [S.l.]: Wikimedia Foundation, 2010. Disponível em: < http://pt.wikipedia.org/wiki/Proxy>. Acesso em: 20 out. 2010.

SAUVÉ, J. P. **Gerência de redes de computadores.** [Campina Grande]:– Paraíba, [2002?]. Disponível em: <http://www.dsc.ufcg.edu.br/~jacques/cursos/2002.1/gr/>. Acesso em: 21 out. 2010.

SILVA, E. dos S. da. **Extensão do modelo de restrições do RBAC para suportar obrigações do modelo ABC**. 2004. 90 f. Dissertação (Mestrado) – Programa de Pós-Graduação em Informática Aplicada, Pontifícia Universidade Católica do Paraná, Curitiba.

UCHÔA, J. Q. **Segurança em Redes e Criptografia**. Lavras: UFLA/FAEPE, 2003. (Curso de Pós Graduação “Lato Sensu” (Especialização) a Distância em Administração em Redes Linux).

UCHÔA, J. Q.; SIMEONE, L. E.; SICA, F. C. **Administração de Redes Linux**. Lavras: UFLA/FAEPE, 2003. (Curso de Pós Graduação “Lato Sensu” (Especialização) a Distância em Administração ao em Redes Linux).

VESPERMAN, J. **Autenticação e o squid**. [S.l.], [2001]. Disponível em: < http://br.geocities.com/cesarakg/AuthenticationAndSquid.html>. Acesso em: 21 out. 2010.

WATANABE, C. S. **Introdução ao cache de web**. [Rio de Janeiro], [2000]. Disponível em: <http://www.rnp.br/newsgen/0003/cache.html>. Acesso em: 20 out. 2010.

ZAGO, A. F. **FAQ:** dicas e indicações de tutoriais sobre webmin, configurador em ambiente gráfico. [S.l.], [2007?]. Disponível em: <http://www.zago.eti.br/webmin.txt>. Acesso em: 24 out. 2010.