UNIVERSIDADE FEDERAL DE SANTA MARIA
COLÉGIO TÉCNICO INDUSTRIAL DE SANTA MARIA
CURSO SUPERIOR DE TECNOLOGIA EM REDES DE COMPUTADORES

GERENCIAMENTO DO *PROXY SQUID* ATRAVÉS DE UMA FERRAMENTA *WEB* COM BASE NA CRIAÇÃO DE PERFIS DE CONTROLE

TRABALHO DE CONCLUSÃO DE CURSO

WILLIAM DRESCH FLORIANO

Santa Maria, RS, Brasil
2016

STRC/UFSM, RS FLORIANO, William Dresch Tecnólogo 2016

GERENCIAMENTO DO *PROXY* SQUID ATRAVÉS DE UMA FERRAMENTA *WEB* COM BASE NA CRIAÇÃO DE PERFIS DE CONTROLE

William Dresch Floriano

Trabalho apresentado ao Curso de Graduação em Tecnologia em Redes de Computadores do Colégio Técnico Industrial de Santa Maria, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de **Tecnólogo em Redes de Computadores.**

Orientador: Prof. Me. Tiago Antonio Rizzetti

Santa Maria, RS, Brasil
2016

Universidade Federal de Santa Maria
Colégio Técnico Industrial de Santa Maria
Curso Superior de Tecnologia em Redes de Computadores

A comissão examinadora, abaixo assinada, aprova o Trabalho de Conclusão de Curso

GERENCIAMENTO DO *PROXY* SQUID ATRAVÉS DE UMA FERRAMENTA *WEB* COM BASE NA CRIAÇÃO DE PERFIS DE CONTROLE

elaborado por
William Dresch Floriano

como requisito parcial para obtenção do grau de
Tecnólogo em Redes de Computadores

Comissão Examinadora

_____
Tiago Antonio Rizzetti, Me.
(Presidente/Orientador)

_____
Tarcisio Ceolin Junior, Me.
(UFSM)

_____
Bolívar Menezes da Silva
(UFSM)

Santa Maria, 08 de julho de 2016

AGRADECIMENTOS

Agradeço primeiramente a minha família, principalmente aos meus pais, Sirley e Jair, aos meus irmãos Suelen e Wagner e a minha cunhada Thais, por todo o suporte e confiança depositados em mim.

A minha namorada Andréa por todo o suporte prestado, incentivando e colaborando para o meu crescimento pessoal e profissional. Certamente é o motivo que torna a minha vida especial.

Ao meu orientador Tiago Antonio Rizzetti, pelos ensinamentos e pela sua disponibilidade, sempre que solicitada.

Aos meus colegas que fizeram parte dessa trajetória acadêmica.

Aos demais professores do curso pelos ensinamentos.

Aos amigos e colegas da empresa Animati, esta que sempre dispôs de flexibilidade e auxílio.

A todos os amigos que, de longe ou perto, estiveram comigo e contribuíram na pessoa que sou hoje.

Muito obrigado a todos.

RESUMO

Monografia
Curso Superior de Tecnologia em Redes de Computadores
Universidade Federal de Santa Maria

GERENCIAMENTO DO *PROXY SQUID* ATRAVÉS DE UMA FERRAMENTA *WEB* COM BASE NA CRIAÇÃO DE PERFIS DE CONTROLE

AUTOR: WILLIAM DRESCH FLORIANO
ORIENTADOR: TIAGO ANTONIO RIZZETTI
Data e Local da Defesa: Santa Maria, 08 de julho de 2016.

O presente trabalho tem como objetivo o desenvolvimento de uma ferramenta *web* que proporcione ao usuário uma nova percepção do uso de softwares de gerenciamento e controle de acesso em redes, sem a necessidade de se ter um grande conhecimento na administração de uma rede. Assim, neste projeto o usuário tem a possibilidade de interagir com o *software SquidGuard,* um módulo aplicado ao *Proxy Squid*. Outro diferencial é o modo de como se realizam os bloqueios. Este se da através da criação de perfis de bloqueio por usuário, onde estarão contidos os sites a serem bloqueados. Utilizou-se como pretexto o controle feito pelo professor quanto ao acesso dos alunos em laboratório. Este por sua vez, gerencia e aplica um perfil de bloqueio e determina em quanto tempo esse irá expirar. Portanto, a ferramenta *web* desenvolvida possibilita o professor gerenciar o controle de acesso do seu laboratório, através do bloqueio de sites indevidos para o período da sua disciplina e disponibilizar somente acesso aos sites necessários.

Palavras-chave: *Proxy Squid*, *SquidGuard*, Controle de Acesso, Gerenciamento, Perfis por Usuário.

LISTA DE ABREVIATURAS E SIGLAS

ACL – *Access Control List*

HTML – *Hyper Text Markup Language*

HTTP – *Hyper Text Transfer Protocol*

HTTPS – *Hyper Text Transfer Protocol Secure*

IP – *Internet Protocol*

LFU – *Least Frequently Used*

LRU – *Least Recent Used*

PHP – *Hypertext PreProcessor*

RFC – *Request For Comments*

TCC – Trabalho de Conclusão de Curso

TCP/IP – *Transmition Control Protocol/Internet Protocol*

UML – *Unifield Modeling Language*

LISTA DE APÊNDICES

LISTA DE IMAGENS

LISTA DE QUADROS

SUMÁRIO

1 INTRODUÇÃO

Vivemos em um mundo globalizado em que praticamente não há obstáculos, isso acontece devido ao advento da Internet, já que, ela possibilita que as pessoas se comuniquem em tempo real, que as empresas vendam seus produtos online e que a transmissão de informações seja muito mais rápida. Assim, o surgimento das novas tecnologias, a cada ano, faz com que o acesso à internet se prolifere cada vez mais e, isto pode ser visto em grandes empresas, afinal, todos utilizam ferramentas vinculadas à rede para melhorar seus serviços, especialmente as redes de computadores. Para Péricas (2003), a confiabilidade de um sistema aumenta com o uso das redes de computadores, afinal, possuem fontes alternativas de fornecimento de dados, dão a possibilidade de escalabilidade da rede e, ainda, ajudam na economia de dinheiro, pois, os computadores pessoais tem custo benefício melhor do que os de grande porte.

Dessa forma, surge um ponto importante na questão do controle de acesso em redes de computadores, especialmente das empresas: estas devem ter um administrador que monitore o controle dos acessos dos usuários na internet. De acordo com Palma e Prates (2000) é necessário que cada empresa tenha um administrador de rede que tenha o poder de fazer o monitoramento e controle dos acessos aos recursos das redes de computadores. Devido a este fato, ferramentas que desempenham funções distintas como o filtro de pacotes e servidores *proxy* foram criadas. Assim sendo, pode-se dizer que o filtro de pacotes atua na camada de rede, assim como os servidores *proxy* agem na camada de aplicação. Ainda, é importante ressaltar que estas camadas foram criadas baseadas no modelo de referência *Transfer Control Protocol - Internet Protocol (*TCP IP). Este foi criado devido ao "desenvolvimento da ARPANET, rede de pesquisa patrocinada pelo Departamento de Defesa dos Estados Unidos" de acordo com Péricas (2003, p. 20 e 35). Entretanto, estes mecanismos só podem ser manipulados por usuários que tenham conhecimento mais avançado.

1.1 Objetivos

O objetivo deste trabalho foi desenvolver uma ferramenta *web*, tendo como base o software livre, que torne possível monitorar e administrar o acesso à internet tendo como meta a criação de perfis de controle individuais. De modo geral, cada perfil irá conter uma lista de

sites desejados para bloqueio, que será aplicada durante um período de tempo, informada pelo criador desse perfil. Desse modo, através da implementação de uma interface intuitiva, o professor poderá criar e gerenciar seus perfis de acesso, apropriado para cada aula, de forma a restringir o conteúdo inapropriado para determinado momento. Quanto aos objetivos específicos propõe-se:

1) Criar e implementar uma interface *web* intuitiva, que possibilite o usuário gerenciar seus perfis, de forma a incluir ou excluir sites para bloqueio;

2) Informar o tempo de aplicação desse perfil e em qual grupo de trabalho;

3) Utilizar *Blacklists* (listas negras) disponíveis na internet e comparar com os sites acessados, informando para o professor a categoria que o site se encaixa;

1.2 Justificativa

A questão do controle de acesso nas redes de computadores é uma alternativa cada vez mais necessária e presente nas instituições de ensino, afinal, torna-se ideal que essas instituições tenham um administrador específico que possa gerir e monitorar o acesso à internet. Porém, em locais como laboratórios de informática, o professor torna-se dependente desse administrador quando se necessita de um controle de acesso mais rígido dos seus alunos.

1.3 Estruturação do trabalho

Este trabalho será dividido em quatro capítulos. No capítulo 2 são explorados os conceitos da parte teórica, que dará um norte para o objetivo a ser cumprido. Conceitos sobre segurança e controle de acesso em rede, que são fundamentados com as explicações de servidor *proxy*, *Squid* e *SquidGuard*, serão tratados. No capítulo 3 é mostrada a proposta deste trabalho, apresentando o modelo de controle e interface a ser implementada pela ferramenta *WEB*.

Além disso, são apresentadas as imagens selecionadas que são importantes para poder identificar a forma pela qual se deu a construção da ferramenta proposta e de como ela deve ser implementada para o administrador saber como deve gerencia-la.

2 FUNDAMENTAÇÃO TEÓRICA

Quando se acessa um site ou uma página *Web* algumas informações relevantes podem ficar armazenadas em logs. Dessa forma, é possível utilizar ferramentas que possibilitem identificar qual computador realizou esse acesso e, caso necessário, restringi-lo por um determinado período de tempo. Ou seja, isto nada mais é do que uma forma de gerenciar o controle de acesso dessas páginas.

Assim, para que o controle de acesso propriamente dito tenha eficácia, surgem mecanismos de segurança, que atuam na execução de códigos internos e seu comportamento é expresso através de modelos de segurança (STALLINGS, 2008).

Portanto, como ferramenta de controle de acesso em páginas *Web*, o presente trabalho se baseia na gerência de redes através da utilização de um servidor *Proxy Squid*. Para isto, torna-se relevante ampliar os conhecimentos e, trazer os conceitos principais que nortearão a pesquisa. Desse modo, o aprofundamento da definição sobre Gerência de Redes, seus aspectos, bem como as subdivisões que nela existem tornam-se primordiais para que o trabalho seja efetivado. A grosso modo, buscar informações sobre a Gerência de Redes fará compreender, da melhor maneira possível, como a ferramenta funcionará.

Após isto, é explorado o conceito de segurança de rede de computadores e, o funcionamento de um servidor *Proxy*, o que levará, diretamente, para outra subdivisão, o trabalho propriamente dito, que é o *Proxy Squid*, filtros, *Proxy* transparente e não transparente, vantagens e desvantagens.

2.1 Segurança e controle de acesso em rede

Falar de segurança, especialmente no que tange às redes de computadores, é crucial atualmente, tanto em sistemas cabeados como em redes sem fio. É um ritual de extrema importância saber se o que está acessando é, de fato, seguro, afinal, pode-se estar correndo algum risco de expor informações privadas. Segundo Wolf e Silva (2011), a segurança é hoje um dos principais problemas que a área da tecnologia da informação vem enfrentando.

Para Tanenbaum (2003), de início, as redes de computadores estavam restritas às pessoas envolvidas com a área da informática. Porém, com o passar do tempo e, graças à evolução tecnológica, as redes de computadores passaram a ser fundamentais no cotidiano de

todo usuários da internet. É possível verificar a existência destas em transações bancárias, por exemplo, e o repasse de informações confidenciais. Este fato faz com que se torne necessário a implantação de alguma ferramenta que restrinja o acesso a determinadas páginas em empresas ou instituições de ensino, por exemplo.

De acordo com Thomas (2007), a elaboração de uma política de segurança é o primeiro passo, e também o principal, para que determinada rede seja segura para o acesso diário. E, é através dessa política de segurança que surgem as ferramentas com regras de acesso às páginas *Web* e, assim que se faça o monitoramento do que está ou não sendo acessado.

Assim, para Wolf e Silva (2011), existem inúmeras tecnologias, bem como ferramentas, que tem como principal meta evitar que esses dados não sejam obtidos de maneira ilícita, como por exemplo, via acesso externo. Entretanto, um mero erro pode ser decorrente de descuido dos acessos internos, que, assim, podem vir a comprometer os níveis de segurança já instaurados.

Esse descuido dos acessos internos fez com que surgisse a necessidade da criação de um controle de acesso, que possibilita monitorar os acessos efetuados pelos usuários e, gerenciá-los conforme a politica de cada instituição. De acordo com Stallings (2008), é através do controle que é feita a limitação do acesso aos sistemas. E, para que isto ocorra, é necessário que toda página, primeiramente, seja identificada ou autenticada antes que esse seja definitivo. Assim, este é passível de configuração para determinado usuário.

A segurança em redes de computadores é, portanto, primordial na era da tecnologia pela qual se está passando e, por isto, é importante falar especificamente de que métodos são mais usados para que esta segurança exista.

2.1.1 Métodos de segurança de redes de computadores

O controle de acesso em uma rede é necessário e, para que este aconteça, torna-se relevante a existência de alguma ferramenta que mantenha o sistema seguro, afinal, uma das principais metas é que a integridade dos dados acessados se mantenha intacta. Por isto, surge então a necessidade de realizar o monitoramento, bem como de limitar o acesso dos usuários na rede. Dessa forma, o acesso ao conteúdo pré-determinado se dará por meio da utilização de um servidor *Proxy*.

2.2 *Proxy*

Quando se acessa um site utiliza-se um navegador *web* que, na grande maioria das vezes, se conecta diretamente com o servidor *web* de destino. Entretanto, há uma diferente conexão, que é através de um servidor *Proxy*. Este atua como uma espécie de intermediação em uma transação *web*. Isto é, de maneira mais clara, um serviço que age entre o computador do usuário e o servidor de destino, ou seja, recebe requisições para que o acesso à Internet seja possível e, assim, busca toda e qualquer informação em seu *cache* e, caso não encontre, busca a requisição no site que é desejado, conforme se observa na figura 1. De acordo com Geib (2008), os proxies costumam ser utilizados em firewalls para manter a segurança. Ainda, eles servem como um transmissor de requisições de determinada rede interna para a Internet.

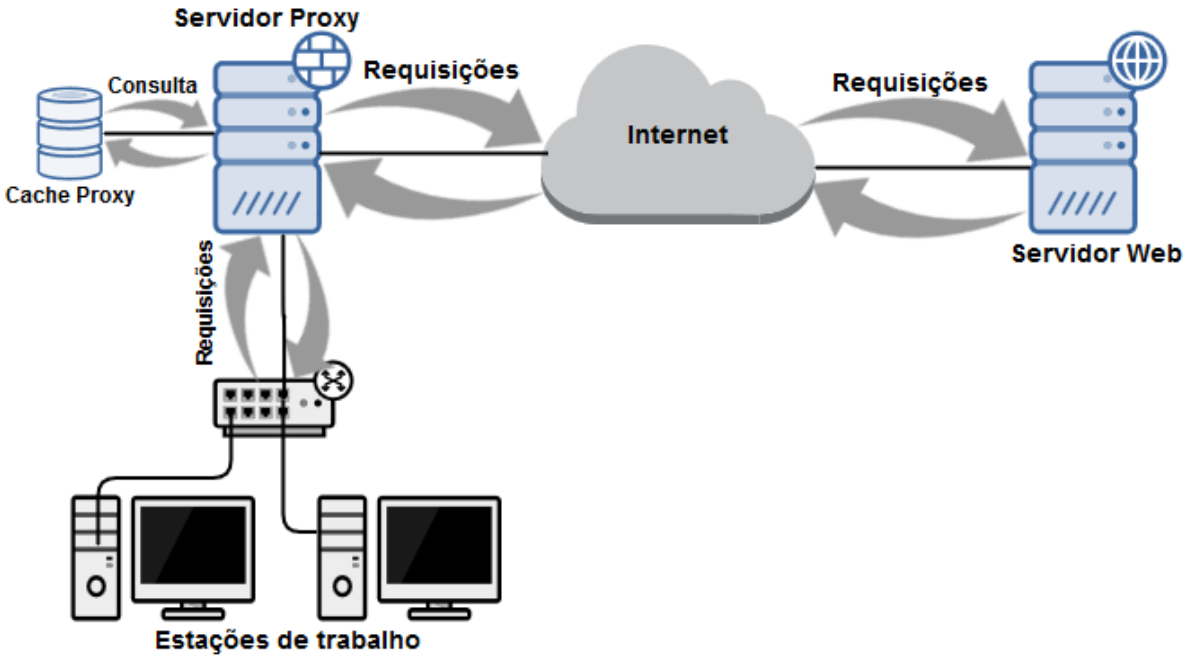

Figura 1 - Controle de requisições pelo Servidor *Proxy*.
Fonte: Acervo Pessoal.

Os proxies também têm um papel fundamental em distintas tarefas. Por exemplo, podem ser utilizados na parte de tradução, varredura antivírus, filtro, controle de acesso e *cache*. Porém, como todo serviço, possui vantagens e desvantagens. Dentre a primeira opção está o isolamento total de redes e balanceamento de carga, que ocorre através da utilização de cache de conteúdo. Já dentre as desvantagens pode-se destacar o fato de que os proxies não

foram feitos para proteger todo o tráfego de internet e sim, para proteger, normalmente, apenas o navegador, e que a configuração deve ser feita separadamente para cada aplicativo - e-mail, por exemplo – e, ainda alguns destes podem não ser compatíveis.

Assim sendo, um servidor *proxy* implantado faz com que possam haver regras para usuários e grupos de usuários. Além disso, é possível também regrar determinados assuntos e conteúdo que possam vir a trafegar na rede, fato que trará maior controle de acesso, de segurança e, ainda, diminuição na latência da rede (SILVA, 2007).

Existem inúmeros tipos de proxies, porém, é importante dar atenção aos que atuam na detecção de tentativas de quebrar a segurança, além, é claro, de possuírem as características principais de um p*roxy*. Devido a isto, este utiliza dois tipos de conexões TCP, sendo uma para o cliente e outra para o servidor.

2.2.1 Filtros do *Proxy*

Uma das características mais importantes de um servidor *proxy* são os filtros. De acordo com Marcelo (2005), estes, atuam por meio de regras, que são pré-determinadas pelo administrador. Para Watanabe (2000), são os administradores que tem a possibilidade de criar regras, sejam elas para filtrar as requisições baseadas no endereço de IP de determinado cliente, da URL, do domínio, das redes, e ainda, do objeto requisitado. Desse modo, a possibilidade de implementar regras pode, então, bloquear requisições ditas como inapropriadas. Um exemplo a ser usado nesta questão são as escolas e organizações que utilizam este método para permitir que determinadas páginas sejam acessadas e outras não. Estas regras, que são denominadas de *Access Control Lists* (ACL), devem ser criadas, segundo Watanabe (2000), baseadas nestes itens: endereço de rede da estação de trabalho; domínio requisitado; rede de origem ou destino; localização do objeto requisitado; período de acesso às páginas de Internet; e habilitar ou não a autenticação.

Esses filtros podem ser usados de duas maneiras: isolados ou em conjunto. Entretanto, sempre são analisados sequencialmente.

2.2.2 *Proxy* Transparente

Um dos recursos de extrema importância do *proxy* é que ele pode ser utilizado de duas maneiras distintas: com transparência ou sem. O *proxy* transparente facilita o uso pelo usuário, afinal, ele não precisará configurar manualmente os *browsers* que serão utilizados para o acesso à Internet. Segundo Marimoto (2009), usar o *proxy* transparente faz com que as solicitações de páginas da Internet sejam interceptadas e, assim, redirecionadas para o *proxy* da rede. Este método nada mais é do que uma forma de dar garantia aos usuários da rede que vão utilizar o servidor.

Além disso, o servidor foi arquitetado para o navegador não saber de sua existência na rede. Ou seja, os navegadores *web*, normalmente, acreditam que estão realizando a comunicação diretamente com o servidor que estão requerendo, quando, na verdade, é o *proxy* que realiza a comunicação, de fato. Ainda, outro fator importante é que o *proxy* transparente não permite filtragem HTTPS, pois quebra a criptografia fim-a-fim, já no não transparente, isso é possível, uma vez que o acesso ao site propriamente dito é realizado somente depois da liberação do endereço pelo *proxy*.

2.2.3 *Proxy* Não Transparente

Este subcapítulo se concentra no conceito de *proxy* não transparente. Conforme desenvolvido na sessão anterior, o *proxy* transparente facilita o acesso à Internet para o usuário, já que, não é necessário configurar um browser manualmente. Em contrapartida, no *proxy* não transparente é preciso configurá-lo manualmente no navegador de cada usuário, ou seja, deverá ser informado qual é o endereço de IP do *proxy* bem como a porta pela qual o mesmo está operando.

Para Duarte (2011), um dos mecanismos que tornam o servidor não transparente melhor que o transparente é que os usuários não precisam, necessariamente, estarem conectados à Internet, e sim, apenas configurados para que as requisições do *proxy* sejam feitas. Dessa maneira, este se encarrega, portanto, de realizar o procedimento final, ou seja, a comunicação propriamente dita.

2.2.4 Vantagens e desvantagens do *Proxy*

De acordo com Watanabe (2000), dentre as principais vantagens estão à redução do tráfego, redução da carga de servidores, redução de latência e possibilidade de acesso.

- **Redução do tráfego:** Trafegam menos requisições na Internet. Assim sendo, o servidor pode recuperar o objeto apenas uma vez e, isto acaba por reduzir a quantidade de banda que é utilizada pelo cliente. É possível conseguir até 60% de taxa de acerto (WATANABE, 2000);

- **Possibilidade de acesso**: Se o servidor WWW de endereço determinado no URL não está acessível ou está recebendo mais solicitações que pode aguentar, será possível acessar a página, desde que esta esteja armazenada no *proxy* (WATANABE, 2000).

Quanto às desvantagens pode-se citar a segurança em protocolos e aplicações, visto que, o servidor *proxy*, apesar de ajudar na segurança de uma rede, não garante que esta seja efetiva com possíveis falhas em protocolos e aplicações. Dessa forma, torna-se ideal que um firewall seja configurado conjuntamente ao *proxy* (MARCELO, 2005).

2.3 Materiais e Métodos

Neste subcapítulo serão tratados conceitos e softwares utilizados para o desenvolvimento desta ferramenta, a qual será utilizada como uma aplicação *web*.

Isto significa que, para que a ferramenta seja desenvolvida, será necessário explorar os conceitos de sistema operacional, características e configuração de *Squid, Squidguard, Blacklists, Squidanalyser,* Servidor *Apache,* HTML, PHP, *Javascript* e *Shell Script.*

Após todos estes conceitos poderá, então, ser discutida a ferramenta *web* proposta.

2.3.1 Sistema Operacional

Para elaboração deste trabalho foi utilizada uma distribuição do sistema operacional GNU/Linux, chamada Debian. Seu princípio parte de uma associação de usuários, denominada Projeto Debian, que têm como principal objetivo a criação de um sistema operacional livre (Debian, 2016). Porém, pode-se utilizar a ferramenta em qualquer outra distribuição baseada nele, devido a compatibilidade na estruturação de arquivos e de configuração utilizada.

2.3.2 *Squid*

Este software surgiu quando, de início, existia apenas o servidor HTTP CERN, que desempenhava a função tanto de atuar como HTTP como quanto de *proxy* e *cache*, este último tendo sido escrito em 1994 por Ari Loutonen.

Em 1994 também foi iniciado o projeto *Harvest* pelo *Internet Research Task Force Group on Resource Discovery*, que se tratava de um conjunto de ferramentas integradas que atuavam na coleta, extração, organização, localização, além de fazer cache e, assim replicar as informações na Internet (GEIB, 2008). No fim de 1994 o pesquisador Duane Wessels juntou-se ao projeto, que começou a se desmembrar em meados de 1995. Assim, segundo Geib (2008), em 1996 Wessels ingressou no *National Laboratory for Applied Network Reseach* para, então, trabalhar em um projeto que tinha como foco principal *cache* que, mais tarde, passou a ser chamado de *Squid*.

Desde esta época o *Squid* evoluiu muito, tanto em tamanho como em funcionalidade. Atualmente, pode-se destacar, por exemplo, o suporte de funções como o redirecionamento de URL, controles de acesso mais elaborados, opções de armazenamento avançadas, modo *surrogate*, interceptação HTTP, *traffic shaping* e distintos módulos para autenticação (GEIB, 2008).

2.3.2.1 Características e funcionamento

O *Squid* é um software compatível com os sistemas operacionais Windows, Linux, FreeBSD, OpenBSD e NetBSD e, segundo Wessels (2004, p. 20) utilizá-lo traz algumas

vantagens, como por exemplo, a redução da carga do servidor *web*. Isto é, o *proxy* funciona como um intermédio entre o usuário e a operação *web*. De acordo com Geib (2008), o proxy realiza o processo de aceitação de requisição do cliente, faz o processamento e, posteriormente, faz um encaminhamento ao servidor *web*. Portanto, assim que o este procedimento for finalizado, há a possibilidade de a requisição ser registrada, rejeitada, ou mesmo modificada antes do encaminhamento final.

No que diz respeito ao cache, o *Squid* atua como uma espécie de armazém, que guarda todos os conteúdos que foram pesquisados da *web* para utilizá-los posteriormente. Segundo Geib (2008), quando o usuário solicitar o mesmo conteúdo, estas poderão ser vista através do cache, já que, não será preciso contatar o servidor *web*.

2.3.2.2 Configurações do *Squid*

De acordo com Marcelo (2005) "o arquivo *squid.conf* é o responsável por todas as configurações. Ali dentro é que serão criadas as listas de acesso (ACLs) e onde poderemos inserir/modificar parâmetros importantes no sistema" (MARCELO, 2005, p. 9).

Desse modo, é no arquivo *squid.conf* em que estão algumas informações, tais como: qual porta de comunicação foi utilizada bem como o endereço de rede do *proxy*; qual configuração deverá ser usada para que seja informado se é um *proxy* cache ou transparente; tamanho da cache utilizada; a rede que está liberada para que seja possível acessar o *proxy*; e tipos de listas de acesso existentes.

Quando o *Squid* é iniciado, o *squid.conf* passa a ser lido de maneira sequencial. Isto significa que as ACLs também são lidas da mesma maneira. Entretanto, se estas listas de acesso acusarem que se trata de um arquivo externo, este, por sua vez, passará a ser analisado no ato em que o *Squid* for iniciado. Caso ocorra alguma alteração, o serviço deverá ser reiniciado para que, assim, configurações novas possam ser aplicadas.

2.3.2.3 *SquidGuard*

O *SquidGuard* é utilizado como uma extensão aplicada ao *Squid*, e tem como principal característica a possibilidade de bloquear um conjunto de listas especificas, classificadas em categorias, denominadas *Blacklists*. Existem algumas listas disponíveis gratuitamente na internet, que são atualizadas constantemente por seus criadores, e que podem ser divididas, quanto ao seu uso, em dois grupos (SquidGuard, 2016):

- Não comercial (exemplos): MESD *blacklists* e Shalla's *Blacklists*. Essa última foi utilizada no presente trabalho, devido a disponibilização de um *shell script*, por parte dos criadores, que trabalha de forma a atualizar, constantemente, as listas utilizadas;
- Comercial (exemplos): Squidblacklist.org e URLBlacklist.com.

No sistema operacional utilizado como base para o desenvolvimento do trabalho, a configuração do *SquidGuard* é feita através do arquivo *squidGuard.conf*, localizado no diretório /etc/squidguard. Esse arquivo contém algumas diretivas como logdir, que indica o local do arquivo de log gerado pelo *SquidGuard* e a dbhome, onde são armazenados os arquivos e as listas utilizadas para bloqueio.

```
  GNU nano 2.2.6          Arquivo: squidGuard.conf

# CONFIG FILE FOR SQUIDGUARD

dbhome /var/lib/squidguard/db
logdir /var/log/squidguard


src admin {
        ip      10.0.1.1-10.0.1.5
}

dest bloqueados {
        domainlist sitesBL
        urllist urlsBL
}

acl {
        admin {
                pass all
        }

        default {
                pass     !bloqueados all
                redirect http://192.168.1.220/bloqueio.php
        }
}

^G Ajuda      ^O Gravar     ^R Ler o Arq ^Y Pág Anter ^K Recort Txt^C Pos Atual
^X Sair       ^J Justificar^W Onde está?^V Próx Pág  ^U Colar Txt ^T Para Spell
```

Figura 2 - Exemplo de configuração do *SquidGuard*.
Fonte: Acervo Pessoal.

Na figura 2 é possível visualizar um exemplo de configuração do *SquidGuard*, que contém a criação de duas classes e duas ACLs. A primeira classe trata-se da origem de endereços IPs dos clientes e lhe é atribuído o nome de “admin”, já a segunda classe, nomeada como “bloqueados”, refere-se aos arquivos que contém domínios e urls a serem bloqueadas. As nomeações das classes facilitam na criação das ACLs, que na figura 2 são definidas como “admin” e “default”. A ACL “admin” significa que todo o acesso está liberado para a range de IPs definida na classe “admin”, e a “default” indica que todo acesso está liberado, exceto os domínios e urls incluídos na classe “bloqueados”.

2.3.2.4 *SquidAnalyser*

O *SquidAnalyser* é um software livre, sob licença *GNU General Public License*, que além de ser disponibilizado para uso, poderá ser modificado perante os termos da licença citada anteriormente (Squidanalyzer, 2014). Sua grande valia se da através da criação de páginas *web*, em virtude da análise dos logs do *Squid*, contendo informações das páginas acessadas pelos usuários, tempo de acesso, sites e urls com maiores acessos, bem como os mais bloqueados, e também uma visualização mais interativa através da demonstração de gráficos, como é possível visualizar na figura 3.

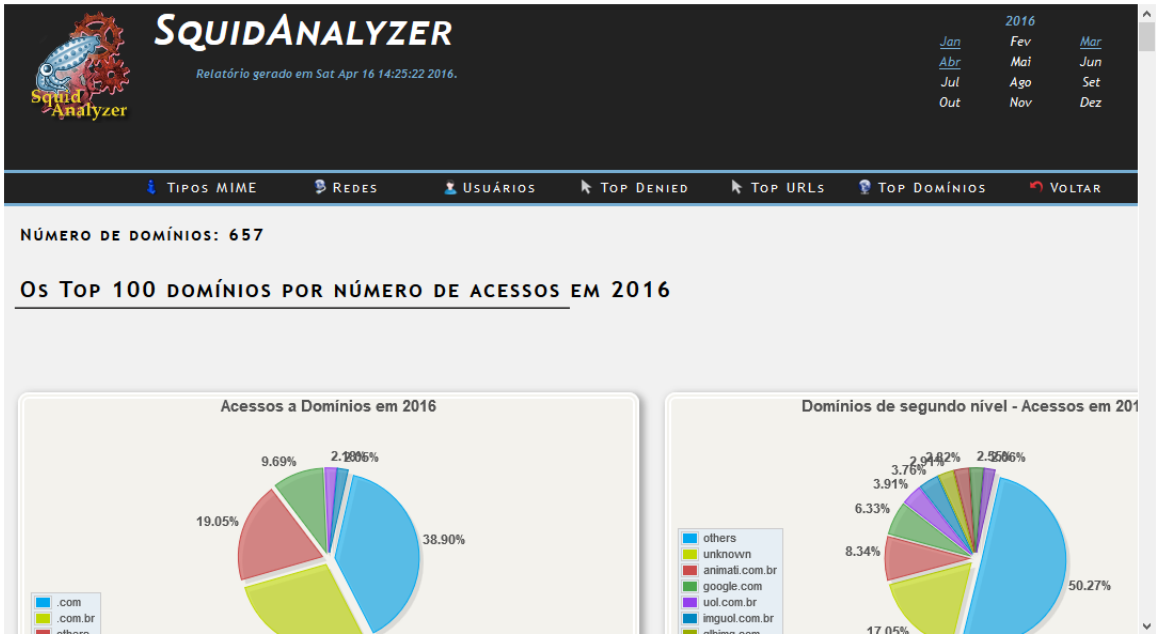

Figura 3 - Software de análise de logs.
Fonte: Acervo Pessoal.

2.3.3 Servidor Apache

O Servidor *Web* Apache, é um dos servidores mais utilizado relacionado à *web*, pois, possui compatibilidade de instalação e, também, é possível que seja utilizado em distintos sistemas operacionais. O Apache é responsável por aceitar os pedidos HTTP dos usuários, ou browser e, após, retorna com uma resposta em HTTP com os objetos pedidos. Ele é largamente utilizado por ser um software livre e, ainda, provê compatibilidade com diversas linguagens de programação como o PHP e Javascript.

3 FERRAMENTAS SIMILARES

Como base de estudo para o desenvolvimento da ferramenta proposta, torna-se necessário o embasamento em ferramentas que possuem o mesmo principio de utilização, ou seja, através da interação do *Proxy Squid* e *SquidGuard* via interface *web*. Desse modo, foram selecionadas duas ferramentas que possuem tal finalidade: Carraro *Dashboard* e *SquidGuard Manager*.

3.1 Carraro Dashboard

O Carraro DashBorad é uma ferramenta *web* que tem como principal objetivo o auxilio na administração dos servidores Linux. Esta administração dos servidores pode ser vista no gerenciamento do *Proxy Squid*, testes de conectividade, NMAP, DHCP, bem como de status de serviços.

Esta ferramenta foi desenvolvida em PHP, que interage com alguns Shell Script, e possui certa facilidade no momento da criação das regras de bloqueio para o *Proxy Squid*. Entretanto, o usuário que fará uso da ferramenta deverá possuir conhecimento específico do assunto, pois, a regra e o grupo em que esta será aplicada deverão ser inseridos manualmente.

Figura 4 - Exemplo de uso da ferramenta Carrarro Dashboard.
Fonte: Acervo Pessoal.

Na figura 4 é possível visualizar a interface de gerenciamento que a ferramenta *web* proporciona, e esta, por sua vez, possui uma licença totalmente Open Source, porém, qualquer alteração que possa vir a auxiliar na correção de erros e falhas do sistema deverá ser reportada ao autor.

3.2 *SquidGuard Manager*

A ferramenta *web SquidGuard Manager* é desenvolvida pelo Gilles Darold e disponibilizada sob os termos *GNU General Public License*. Nela é possível configurar e gerenciar o software *SquidGuard* através da leitura de seu arquivo de configuração, o *squidGuard.conf*. Dentre algumas de suas funcionalidades podemos destacar a configuração de datas e horários para bloqueio de listas, redirecionamentos de urls desejadas, criação e aplicação de filtros de controles através da utilização de listas cadastradas, gerenciamento dessas listas por meio de inclusão ou exclusão de domínios e urls e a edição ou inserção de novas ACLs.

Por fim, após modificar qualquer configuração disponível na ferramenta, será necessário reiniciar o *Squid* e, para isso, é disponibilizada uma opção no menu chamada "*Restart Squid*". Na figura 5 é possível visualizar a interface *web SquidGuard Manager*.

Figura 5 - Interface da ferramenta *web* SquidGuard Manager.
Fonte: Acervo Pessoal.

3.3 Comparativo entre as ferramentas similares e a proposta

Após a descrição das ferramentas Carraro Dasboard e *SquidGuard Manager* acima é possível perceber que a primeira possui uma facilidade de manuseio maior para o usuário, mesmo que este possua pouco conhecimento no assunto. Já a segunda, exige uma configuração mais detalhada das funcionalidades, requisitando instruções mais aprofundadas no assunto.

Tendo em vista que as duas ferramentas citadas necessitam de diversas configurações manuais e uma delas exige um aprofundamento demasiado na temática, surgiu a necessidade de desenvolver uma ferramenta mais dinâmica e intuitiva, de forma que possibilite o utilizador criar e gerenciar um perfil de controle de acesso e aplica-lo para grupos pré-definidos. Assim, torna-se mais fácil de manuseá-lo, independente do nível de conhecimento técnico para tal operação, pois, serão sugeridas opções de bloqueio para inserir em seu perfil gerenciável. No quadro 1, é possível visualizar as características das ferramentas já existentes, bem como o diferencial do trabalho proposto.

Características	Carraro DashBoard	SquidGuard Manager	Ferramenta Proposta
Facilidade quanto ao uso por pessoas com pouco conhecimento técnico	Ferramenta possuí configuração manual do *Proxy Squid* e algumas funcionalidades aplicadas a servidores, desse modo, é necessário ter certo cuidado na aplicação dessas configurações.	Ferramenta desenvolvida e voltada para o gerenciamento de diversos softwares, não possui uma interface intuitiva e de fácil utilização.	Ferramenta desenvolvida e pré-configurada com grupos de bloqueios, sem a necessidade de o utilizador configurar regras manualmente.
Propõe a atribuição de regras por grupos específicos de usuários	Grupos e regras pré-determinados, sendo possível alterá-las quando necessário.	Consegue-se implementar regras para determinados grupos e alterá-las quando for necessário.	Grupos e regras pré-determinados, sendo possível gerenciá-los através da criação e aplicação de perfis individuais.
Sugestões de bloqueio através de uma listagem automática de sites acessados por grupo de usuários	Não possui tal funcionalidade devido à necessidade de inserções de bloqueios manuais.	Não possui tal funcionalidade devido à necessidade de configurar manualmente as listas de bloqueios.	Possui tal funcionalidade e aplica-se de forma automática para o usuário.
Utilização de *Blacklists* e atualização automática destas	Não possui um modo de configuração para *Blacklists*.	Pode-se configurar manualmente a utilização de *Blacklists*.	Pode-se aplicar um conjunto de *Blacklists* para um grupo em específico e estas são atualizadas diariamente através de um *shell script*.

Quadro 2 – Comparativo entre as ferramentas similares e a proposta.

4 DESCRIÇÃO DO SISTEMA

4.1 Modo de utilização

O método selecionado para controlar o acesso à rede externa foi o *proxy* não transparente, ou seja, sua configuração é feita de forma manual no navegador, necessitando a inserção dos dados referentes ao *proxy* que será utilizado. A escolha do *proxy* não transparente deve-se a possibilidade de bloquear conexões HTTPS, caso ocorram. A figura 6 possibilita visualizar a configuração necessária.

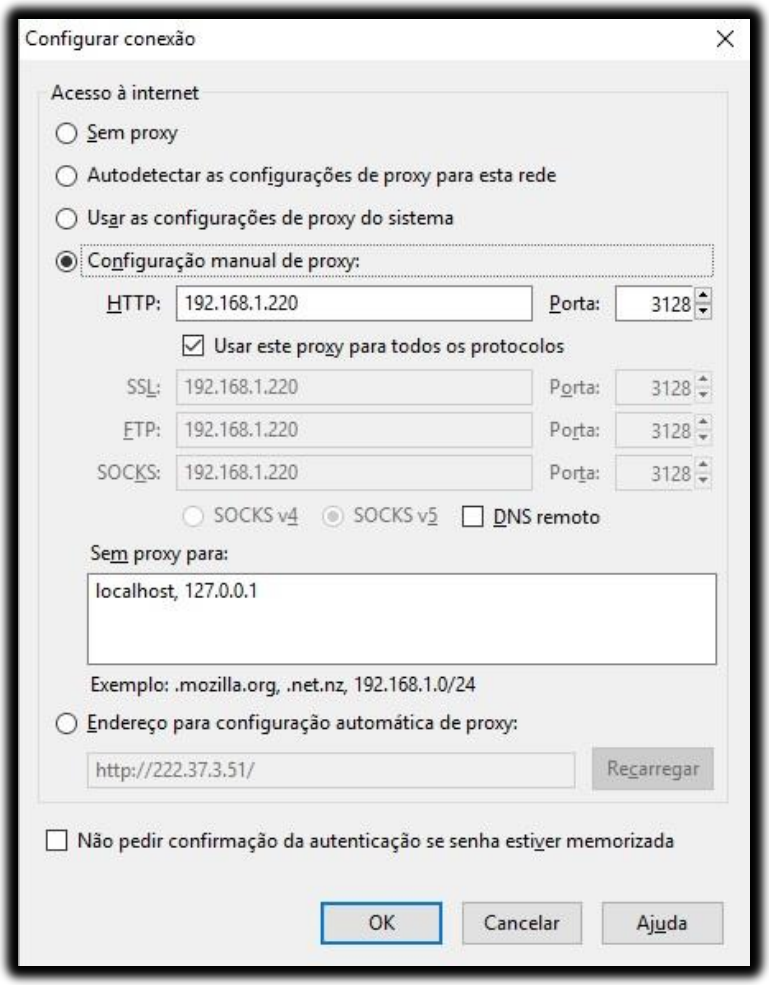

Figura 6 - Configuração do *Proxy* no navegador do usuário.
Fonte: Acervo Pessoal.

4.2 Configurações de acesso

Por padrão, o servidor *proxy Squid* foi configurado de maneira a liberar todo e qualquer acesso. Sendo assim, o controle desse acesso será feito de forma individual para cada laboratório, tendo como base o perfil selecionado pelo professor. Uma vez que determinado usuário acessar um site, este será adicionado ao arquivo de acessos do laboratório, contendo o site acessado e sua categoria. A categoria indicada será o retorno da comparação do site acessado com o conjunto de *Blacklists* utilizadas pelo *proxy Squid*. Desse modo, é possível gerenciar individualmente o perfil aplicado ao laboratório, possibilitando a inclusão ou exclusão desse site para bloqueio.

A figura 7 possibilita compreender o modo como o professor poderá gerenciar seu perfil, de acordo com os bloqueios e liberações que efetuar, bem como, bloquear *blacklists* para o laboratório selecionado, e também sendo possível visualizar um relatório completo dos acessos ao servidor *proxy Squid*.

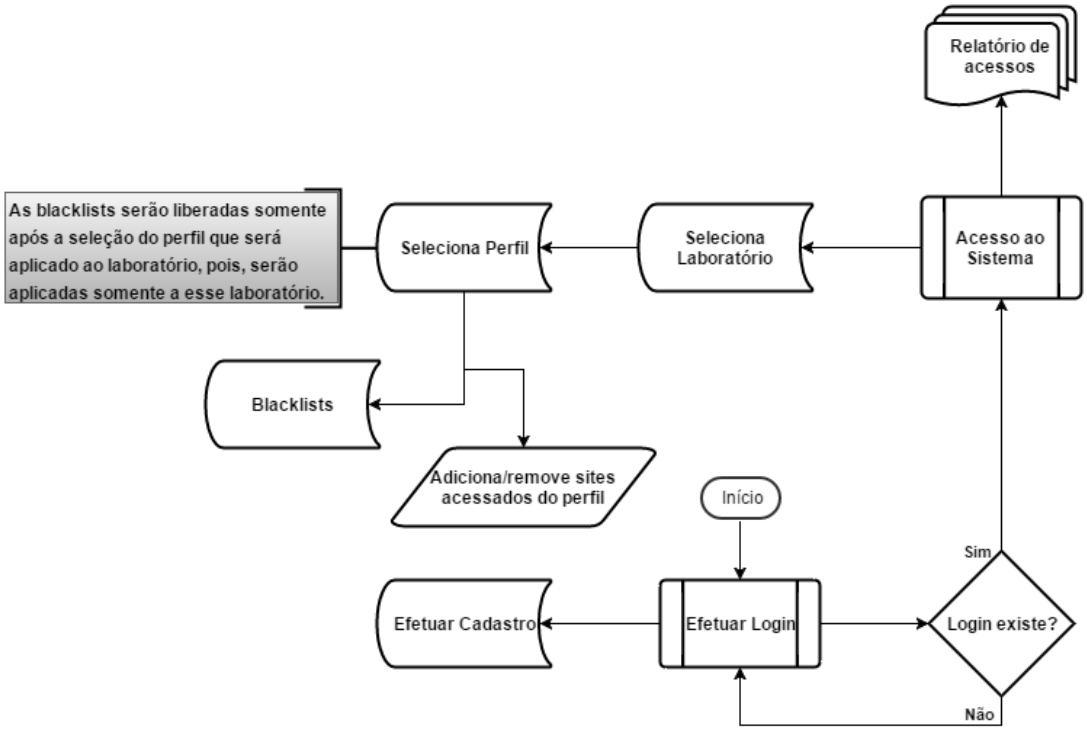

Figura 7 - Gerenciamento de laboratório a partir dos perfis de acesso.
Fonte: Acervo Pessoal.

Para melhor entendimento da figura 7, serão apresentados casos de uso, utilizando diagramas de *Unifield Modeling Languade* (UML).

4.3 Sistema de banco de dados

O sistema de banco de dados surgiu da necessidade de obter um controle mais efetivo dos dados, e garantir a integridade dos mesmos. Desse modo, este sistema irá verificar os dados inseridos no banco, as ligações entre as tabelas, através de chaves estrangeiras, e as informações referentes aos usuários cadastrados. Na figura 8 observa-se a estrutura do banco de dados utilizado para esse trabalho.

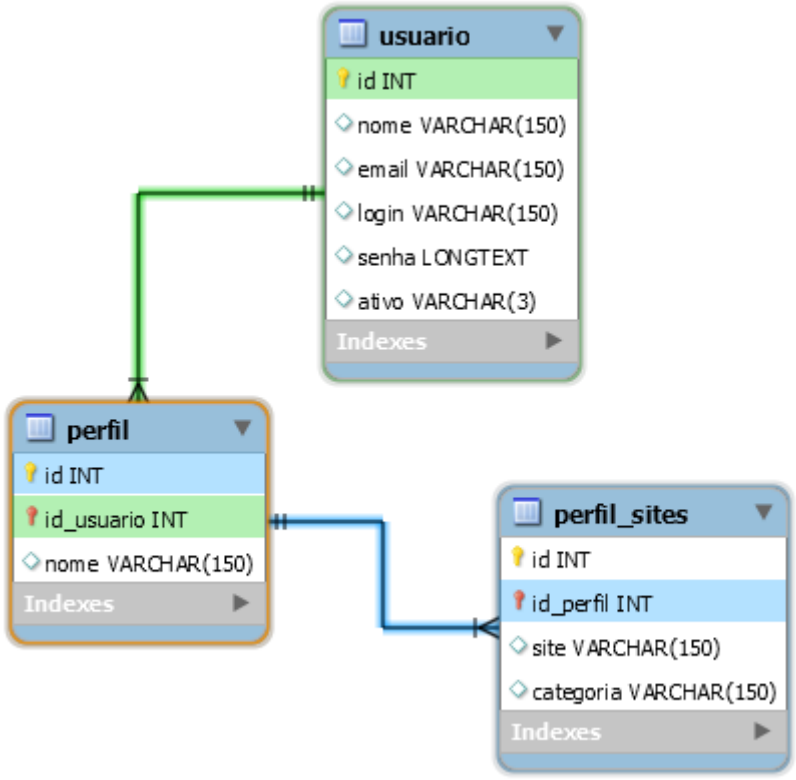

Figura 8 - Tabelas do sistema de banco de dados.
Fonte: Acervo Pessoal.

Descrição das tabelas:

- Tabela usuário: possui informações dos usuários cadastrados no sistema e um campo, chamado “ativo”, que determina se o usuário possui, ou não, permissão de acesso ao sistema;

- Tabela perfil: possui o nome dos perfis cadastrados pelos usuários e um identificador que relaciona o perfil ao usuário, pois, um usuário pode ter vários perfis, mas um perfil está associado somente a um usuário;

- Tabela perfil_sites: contém os sites selecionados para bloqueio e aplicados ao perfil do usuário, juntamente com a sua categoria. Também possui um identificador que relaciona os sites bloqueados ao perfil selecionado.

4.4 Estrutura dos arquivos de configuração

A escolha do modo de controle das informações obtidas e fornecidas ao *proxy squid*, foi definida através da verificação e edição de seus arquivos de configuração. Essa busca e inserção de dados se da através da execução de *shell scripts*, em consequência das chamadas de sistema executadas pela interface *web*. É possível observar no quadro 2, a divisão desses arquivos em dois grupos de controle, geral e individual, e um terceiro grupo, responsável pela criação da tabela do laboratório selecionado, o qual possui um padrão de configuração visível na interface *web*, o *datatable*.

Arquivos de configurações (grupos)		
Geral (*squidguard*)	Individual (Perfil)	Datatable
Lab+(numero_do_lab)	Lab+(numero_do_lab).txt	allLab+(numero_do_lab)
blbloqLab+(numero_do_lab)	Lab+(numero_do_lab)time.txt	blackbloqLab+(numero_dolab)
bllibLab+(numero_do_lab)		blacklibLab+(numero_dolab)

Quadro 2 - Arquivos de configurações.

Seguem abaixo as informações contidas nos arquivos de configurações:

- Arquivo Lab+(numero_do_lab): contém os sites bloqueados para o laboratório selecionado. Esse arquivo é modificado, automaticamente, conforme as alterações feitas no perfil aplicado ao laboratório, e aplica-se a todos os usuários que estiverem utilizando o laboratório selecionado.

- Arquivo blbloqLab+(numero_do_lab): contém um conjunto de *blacklists* que estão bloqueadas para acesso e são aplicadas ao laboratório selecionado. É utilizado para indicar tal definição ao arquivo de configuração do *squidguard* e aplicam-se a todos os usuários que estiverem utilizando o laboratório selecionado.

- Arquivo bllibLab+(numero_do_lab): contém um conjunto de *blacklists* que estão liberadas para acesso e são aplicadas ao laboratório selecionado. É utilizado para indicar tal definição ao arquivo de configuração do *squidguard* e aplicam-se a todos os usuários que estiverem utilizando o laboratório selecionado.

- Arquivo Lab+(numero_do_lab).txt: contém os sites bloqueados em relação ao perfil que está sendo aplicado ao laboratório selecionado. Esse arquivo é modificado, automaticamente, conforme as alterações feitas no perfil aplicado ao laboratório.

- Arquivo Lab+(numero_do_lab)time.txt: contém o tempo que o perfil selecionado será aplicado no laboratório. Esse tempo servirá para expirar as configurações aplicadas ao laboratório selecionado, como por exemplo, perfis e *blacklists*.

- Arquivos *datatable*: de modo geral, esses arquivos contém um determinado padrão, que possibilita a visualização dos seus dados através de uma tabela na interface *web*. O padrão determina que o conteúdo contido entre aspas (“”) representa uma coluna da tabela gerada, exceto o “*data*”. As figuras 10 e 11 representam isso de uma melhor forma.

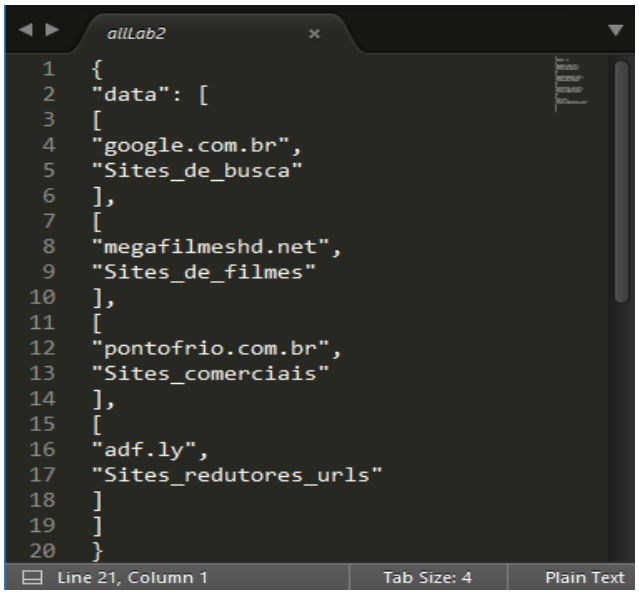

Figura 9 - Formato arquivo datatable.
Fonte: Acervo Pessoal.

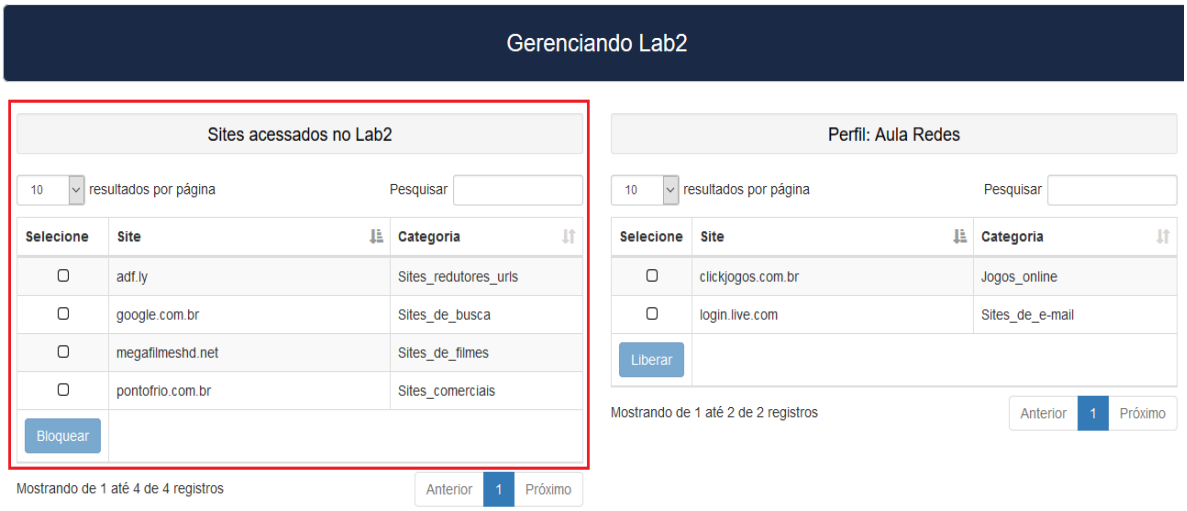

Figura 10 - Tabela gerada com as informações do arquivo da figura 9.
Fonte: Acervo Pessoal.

Para que fosse possível atualizar constantemente a tabela visualizada na figura 10, foi desenvolvido um *shell script* que faz a leitura do *access.log* do *squid*, a cada dois minutos, e insere as informações dos sites acessados, referente ao laboratório selecionado. Na figura 11 é possível visualizar um fragmento do código desenvolvido, e no apêndice I o código completo com comentários que auxiliam na compreensão.

ger_lab_table

```
99
100 #verifica em qual range de ips o usuario se encontra
101 #para inserir o site no arquiv allLab* correto
102 name_lab=`awk 'BEGIN {
103         if (("'"$user"'") >= "192.168.1.100" && ("'"$user"'") <= "192.168.1.120") print ("Lab1");
104         else if (("'"$user"'") >= "192.168.1.121" && ("'"$user"'") <= "192.168.1.200") print ("Lab2");
105         else if (("'"$user"'") >= "192.168.1.201" && ("'"$user"'") <= "192.168.1.250") print ("Lab3");
106         }'`
107
108       #verifica se o site esta no perfil selecionado para o Lab*
109       #no arquivo /var/www/tcc/arquivos/labs/Lab*.txt
110       #se nao estiver ele verifica se ja consta no arquivo allLab*
111         grep -x "$domain" "$perfil_lab$name_lab.txt" >> /dev/null
112         if [ $? -eq 1 ]
113           then
114             #site nao esta no arquivo de perfil que esta sendo utilizado
115          grep -x "\"$domain\"," "$all_sites_lab$name_lab" >> /dev/null
116          if [ $? -eq 1 ]
117            then
118              #site nao esta no arquivo allLab*
119              #sera adicionado ao arquivo allLab*
120              sed -i '3i\[\n"'$domain'",\n"'$categ_site'"\n],' "$all_sites_lab$name_lab"
121          fi
122         else
123            #site esta no perfil utilizado no Lab*
124            #verifica se o site está no arquivo do allLab*
125            #se estiver ele remove, se não, não faz nada
126            var=`grep -nx "\"$domain\"," "$all_sites_lab$name_lab" | cut -d: -f1`
127               if [ "$var" != "" ]
128                then
129                 sed -i ''$((var - 1))','$((var + 2))'d' "$all_sites_lab$name_lab"
130                fi
          fi
```

Figura 11 - Fragmento do shell script utilizado para geração das tabelas de acesso dos laboratórios.
Fonte: Acervo Pessoal.

4.5 Caso de uso do acesso usuário ao *Squid*

Com o objetivo de compreender e visualizar a integração da interface *web* com o servidor *proxy squid*, serão descritos os casos de uso de acesso para o aluno e o professor.

4.5.1 Caso de uso usuário Aluno

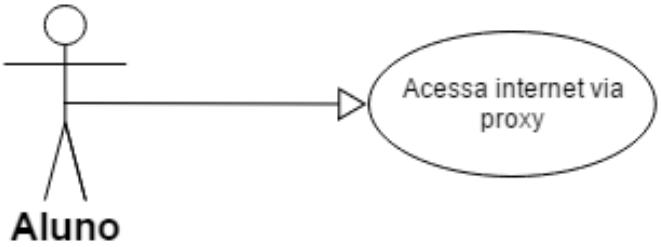

Figura 12 - Diagrama de caso de uso do acesso do Aluno.
Fonte: Acervo Pessoal.

Para melhor compreensão da figura 12, é possível visualizar o apêndice A.

4.5.2 Caso de uso usuário Professor

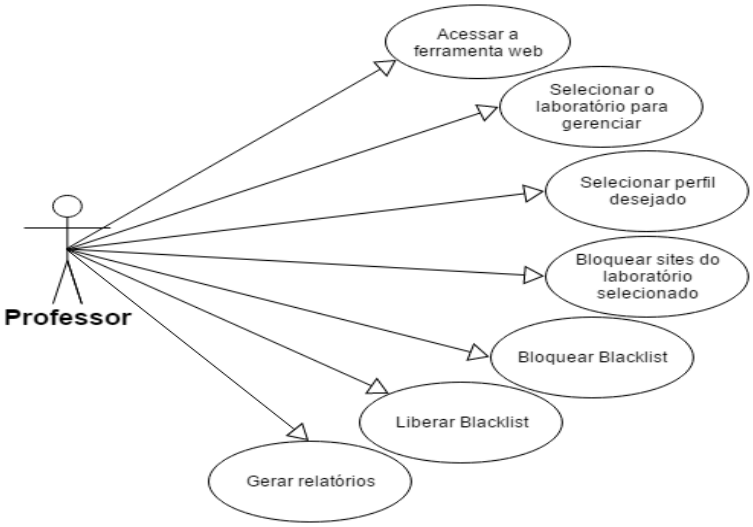

Figura 13 - Diagrama de caso de uso do Professor.
Fonte: Acervo Pessoal.

Para melhor compreensão da figura 13, é possível visualizar o apêndice B até o apêndice H.

4.6 Descrição da ferramenta *Web*

Através da ferramenta *web*, será possível efetuar um gerenciamento mais direto e efetivo do *proxy squid*, independente do nível de experiência que o administrador possui sobre os princípios básicos de funcionamento desse software. A interface foi desenvolvida em HTML, PHP e JavaScript, possibilitando que o professor visualize suas ações de forma mais intuitiva, através dos avisos sobre as operações efetuadas.

4.6.1 Página de login

A primeira página a ser visualizada, será a de login, ou seja, para dispor das funcionalidades fornecidas pela ferramenta *web*, o professor deverá ter efetuado o login nessa página e consequentemente será redirecionado para a página inicial do sistema.

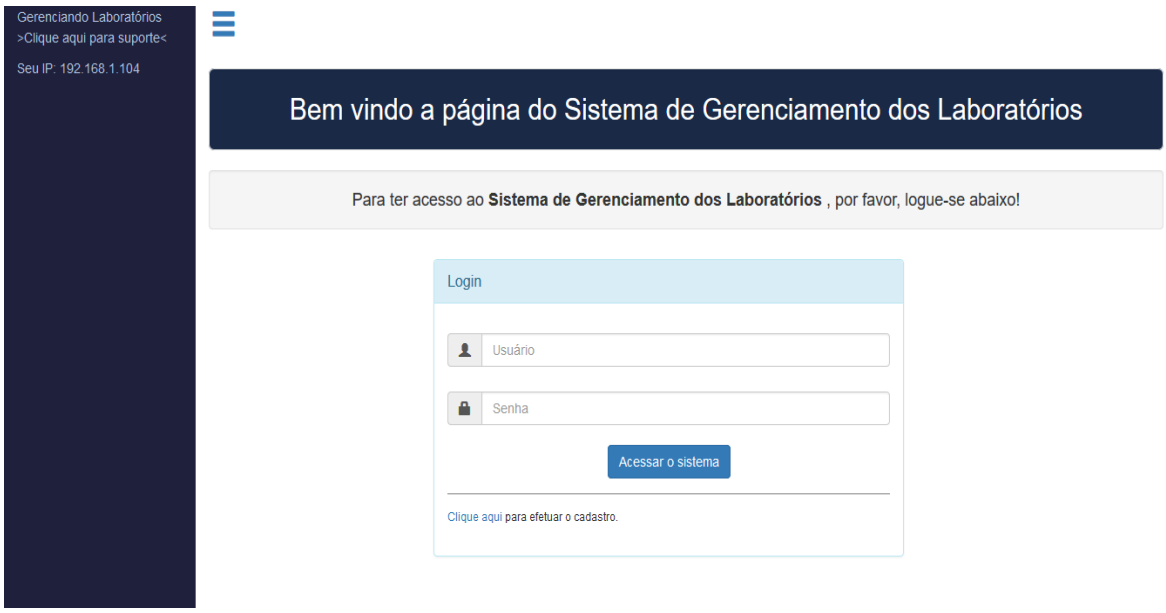

Figura 14 - Pagina de login da ferramenta *web*.
Fonte: Acervo Pessoal.

É possível observar na figura 14 um link para efetuar cadastro, caso o professor não possua. Se o professor desejar fazer o cadastro, ele irá clicar nesse link e visualizará um formulário para preencher suas informações, conforme consta na figura 15.

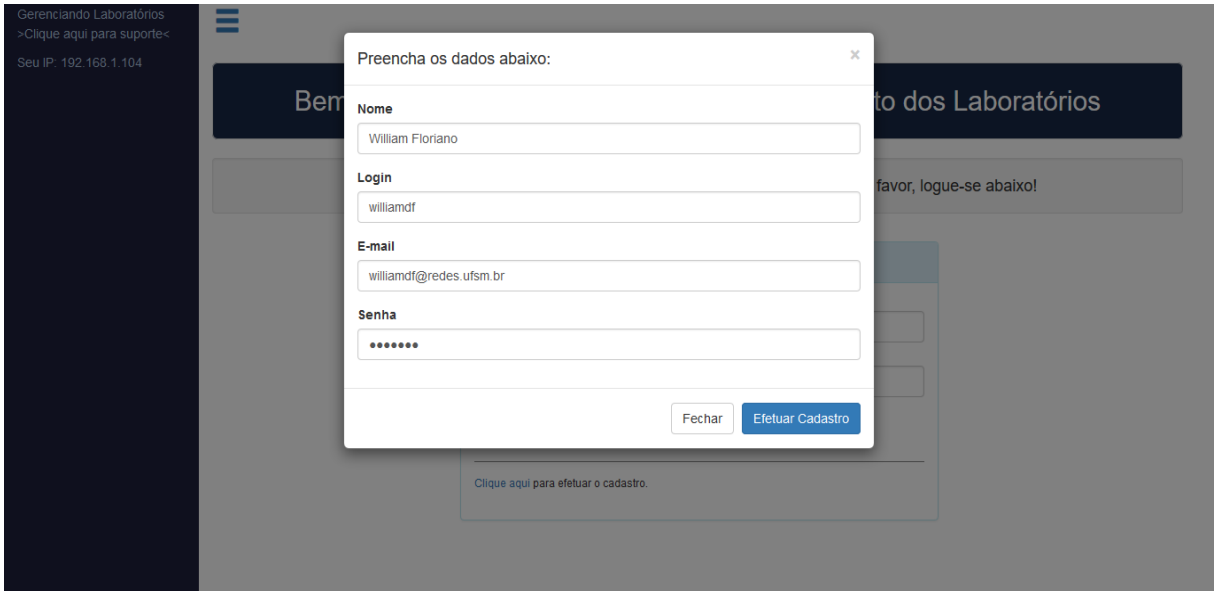

Figura 15 - Formulário de cadastro da ferramenta *web*.
Fonte: Acervo Pessoal.

De modo a tornar mais seguro o armazenamento das senhas cadastradas pelos utilizadores da ferramenta, aplicou-se a criptografia SHA-1 (*Secure Hash Algorithm* – Algoritmo de dispersão seguro) nessas senhas, podendo ser observado na figura 16 o código utilizado para tal.

cadastraUsuario.php

```
26
27  $recebeSenha = filter_input(INPUT_POST, 'senha', FILTER_SANITIZE_SPECIAL_CHARS);
28
29  //Função para criptografar a senha
30  function criptoSenha($criptoSenha){
31  return sha1(md5($criptoSenha));
32  }
33  //Aqui realizo a criptografia da senha
34  $criptoSenha = criptoSenha(filter_input(INPUT_POST, 'senha', FILTER_SANITIZE_SPECIAL_CHARS));
35
```

Figura 16 - Código utilizado para criptografar a senha antes de inserir no banco de dados.
Fonte: Acervo Pessoal.

4.6.2 Página inicial

A página inicial do sistema contém um breve roteiro de utilização da ferramenta *web*, informando o conteúdo existente em cada opção do menu lateral.

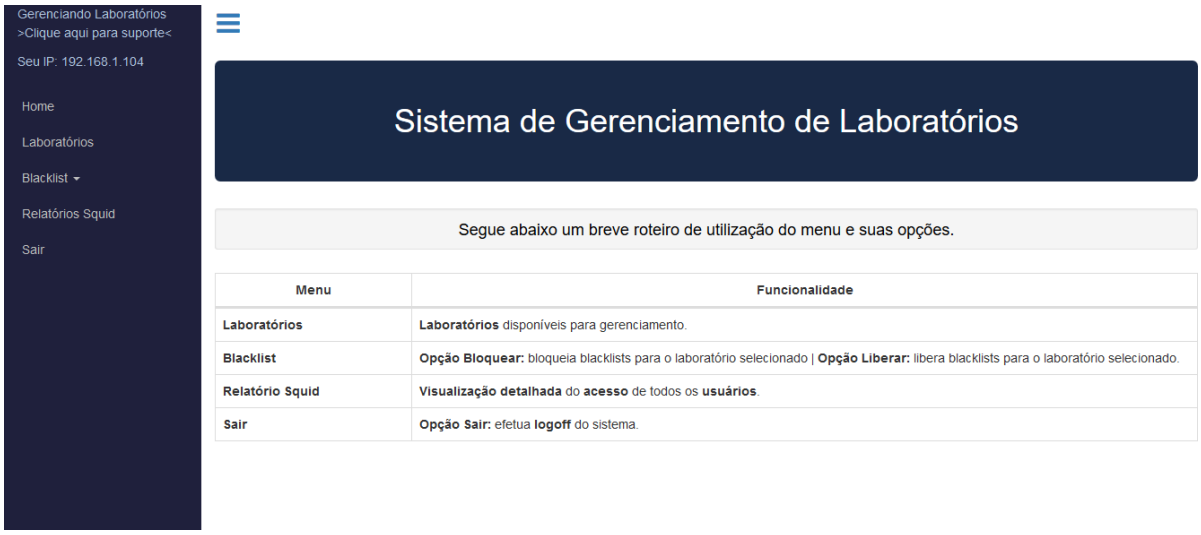

Figura 17 - Página inicial da ferramenta *web*.
Fonte: Acervo Pessoal.

O acesso à interface não necessita ser somente pelo computador, pois, ela se adapta aos dispositivos que requisitarão seus serviços, tornando-se uma interface responsiva. Dessa forma, é possível utilizá-la através de dispositivos móveis.

4.6.3 Lista de laboratórios

Nessa opção do menu é possível verificar os laboratórios que estão disponíveis para gerenciamento. A lista de laboratórios é pré-cadastrada pelo administrador do servidor, pois, cada laboratório possui uma range de IPs e estes deverão ser configurados no *squidguard*, para que o bloqueio funcione corretamente.

Figura 18 - Lista de laboratórios disponíveis.
Fonte: Acervo Pessoal.

Após ocorrer a seleção do laboratório por parte do professor, surgirá uma nova página. Esta por sua vez, possuirá campo para que seja cadastrado um novo perfil, como se observa na figura 19.

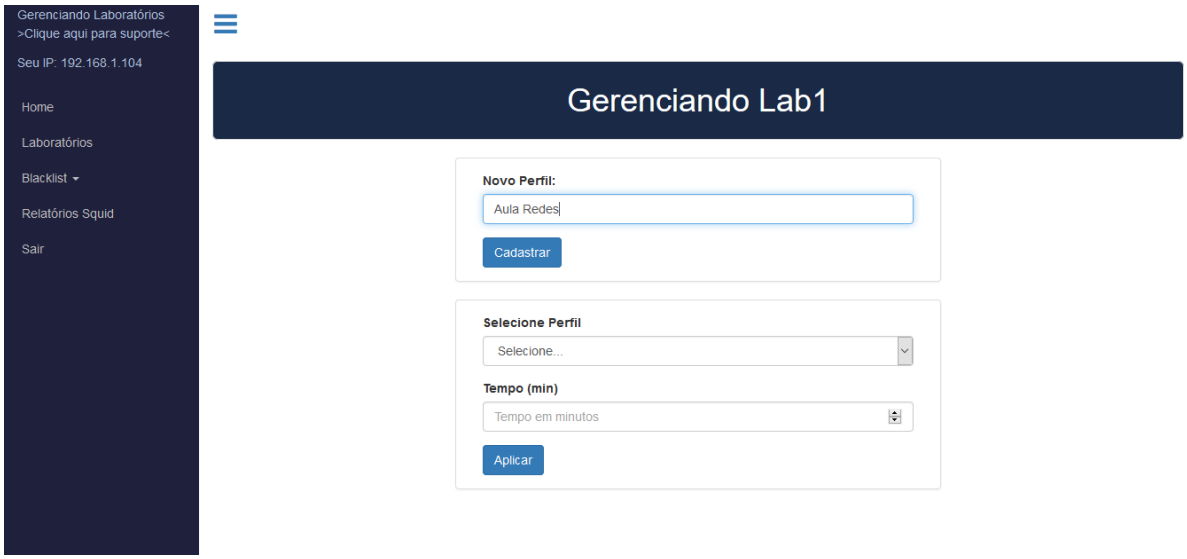

Figura 19 - Cadastro de perfil de bloqueio.
Fonte: Acervo Pessoal.

Caso o professor possua um perfil cadastrado anteriormente, ele poderá selecioná-lo conforme a figura 20.

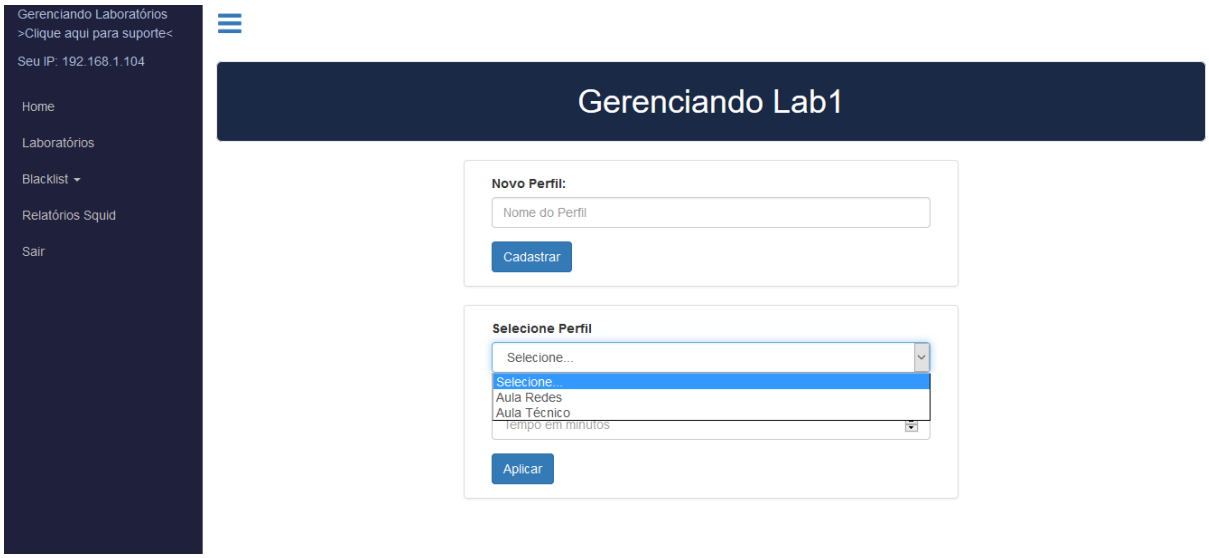

Figura 20 - Seleção de perfis de bloqueio cadastrados.
Fonte: Acervo Pessoal.

Para que o perfil seja aplicado ao laboratório, deverá ser informado um tempo de duração, como se pode observar na figura 21. Esse tempo implicará no bloqueio dos sites inseridos no perfil para o laboratório selecionado. Após esse período de tempo, as regras de bloqueios aplicadas a esse laboratório, serão zeradas.

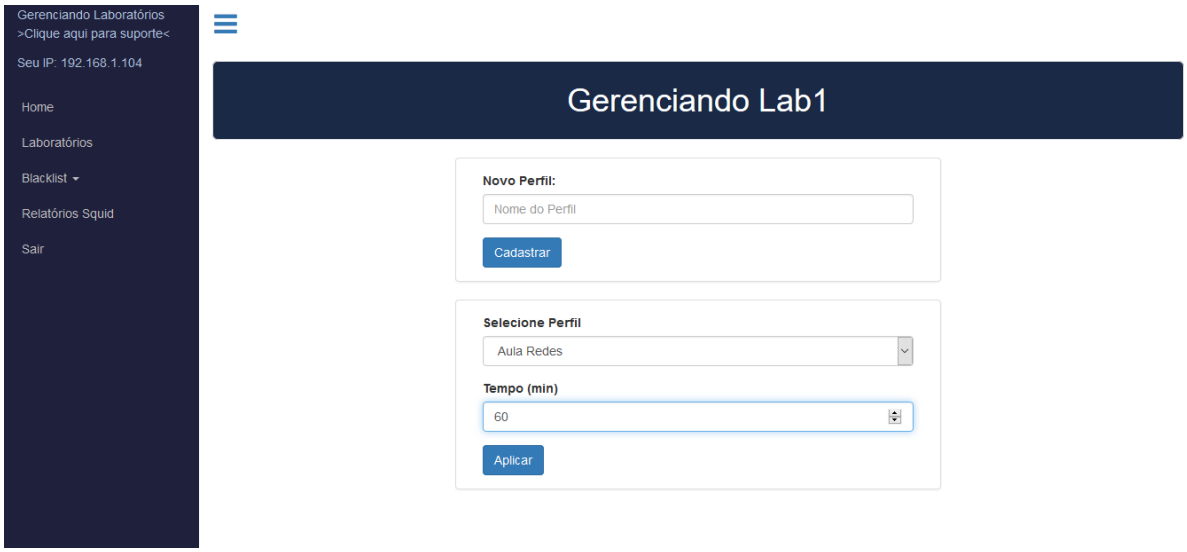

Figura 21 - Inserindo tempo de bloqueio para o perfil selecionado.
Fonte: Acervo Pessoal.

Após selecionar o perfil e inserir o tempo de sua aplicação, serão listados todos os sites acessados no laboratório na data corrente, conforme é possível visualizar na figura 22. Assim, quando se deseja bloquear um ou mais sites, basta selecioná-los e clicar em bloquear.

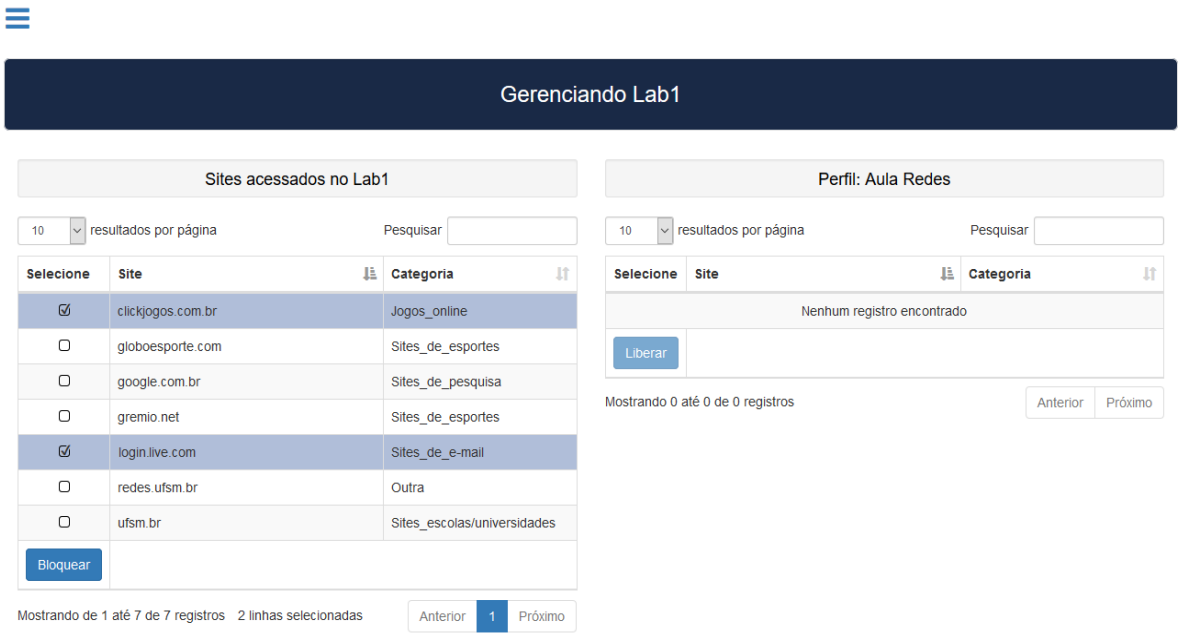

Figura 22 - Sites acessados no laboratório selecionado e perfil selecionado para bloqueio.
Fonte: Acervo Pessoal.

Feita a seleção dos sites e a inserção dos mesmos ao perfil aplicado, através do botão bloquear, será visualizada uma informação que os sites foram bloqueados, e estes serão adicionados automaticamente para bloqueio no laboratório selecionado.

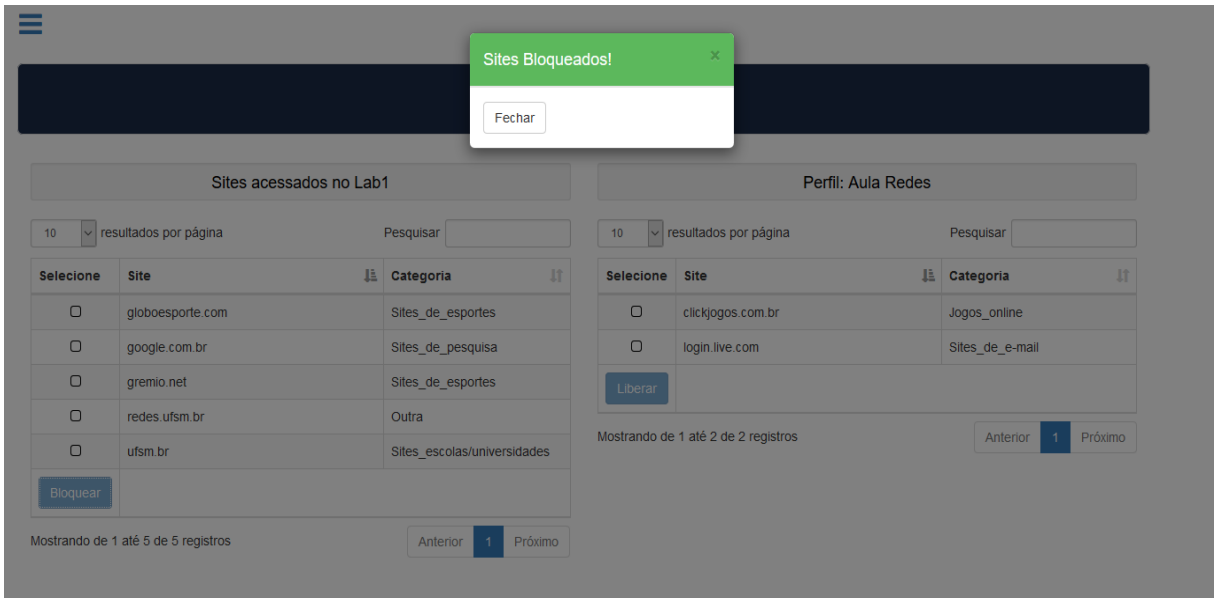

Figura 23 - Mensagem de sucesso no bloqueio dos sites e inserção no perfil.
Fonte: Acervo Pessoal.

4.6.4 *Blacklist*

Ao selecionar essa opção do menu, serão listadas duas opções: “Bloquear” e “Liberar”. A primeira opção trata-se da listagem de *blacklists* liberadas para o laboratório selecionado, assim, selecionando uma ou mais *blacklists* para bloqueio, acarretará em um bloqueio geral dos domínios e urls contidos nela, como se visualiza na figura 24.

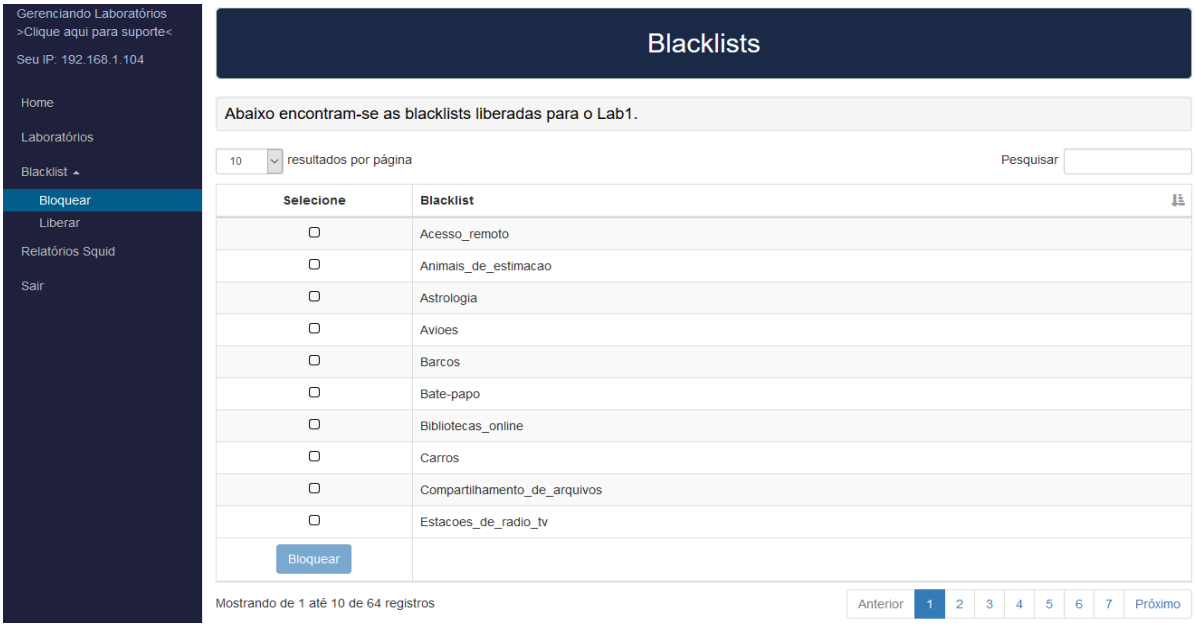

Figura 24 - Bloquear blacklist(s) para o laboratório selecionado.
Fonte: Acervo Pessoal.

Por sua vez, a segunda opção possibilitará visualizar as *blakclists* bloqueadas para o laboratório em questão, bem como a opção de liberar qualquer *blacklist* ali citada, conforme se pode visualizar na figura 25.

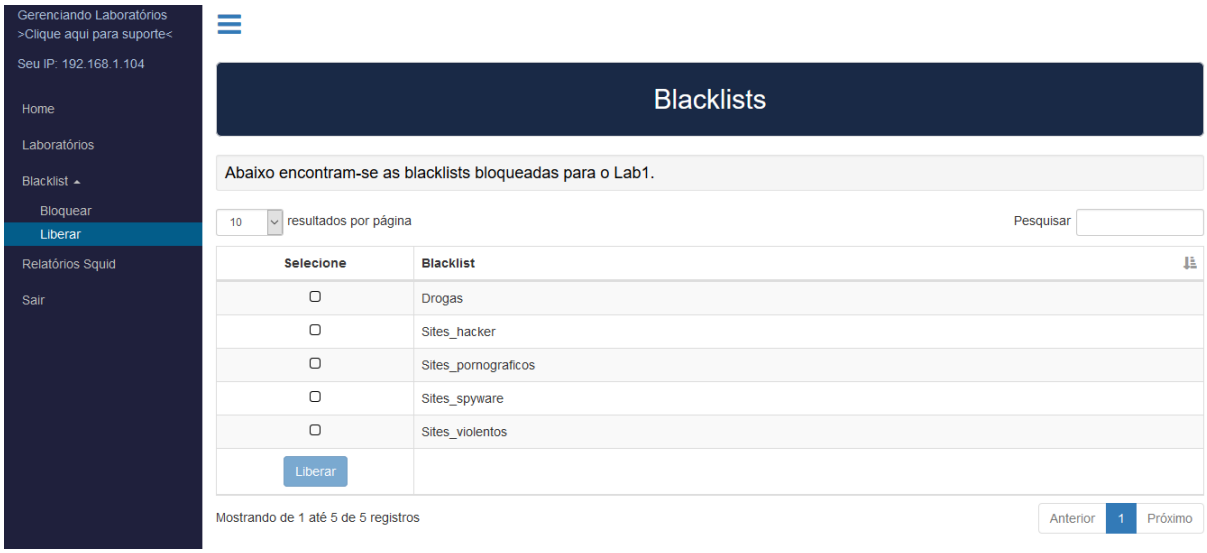

Figura 25 - Liberar blacklist(s) para o laboratório selecionado.
Fonte: Acervo Pessoal.

4.6.5 Relatórios *Squid*

O *squid* possui um arquivo de registro dos acessos de cada usuário, chamado *access.log*. Cada acesso, ou tentativa do mesmo, é registrado nesse arquivo, bem como os domínios e urls acessados. A figura 23 representa uma visualização mais amigável desse arquivo de *log* através da utilização do software *squidanalyzer*, disponibilizando as informações através de gráficos e estatísticas de uso do *proxy squid*.

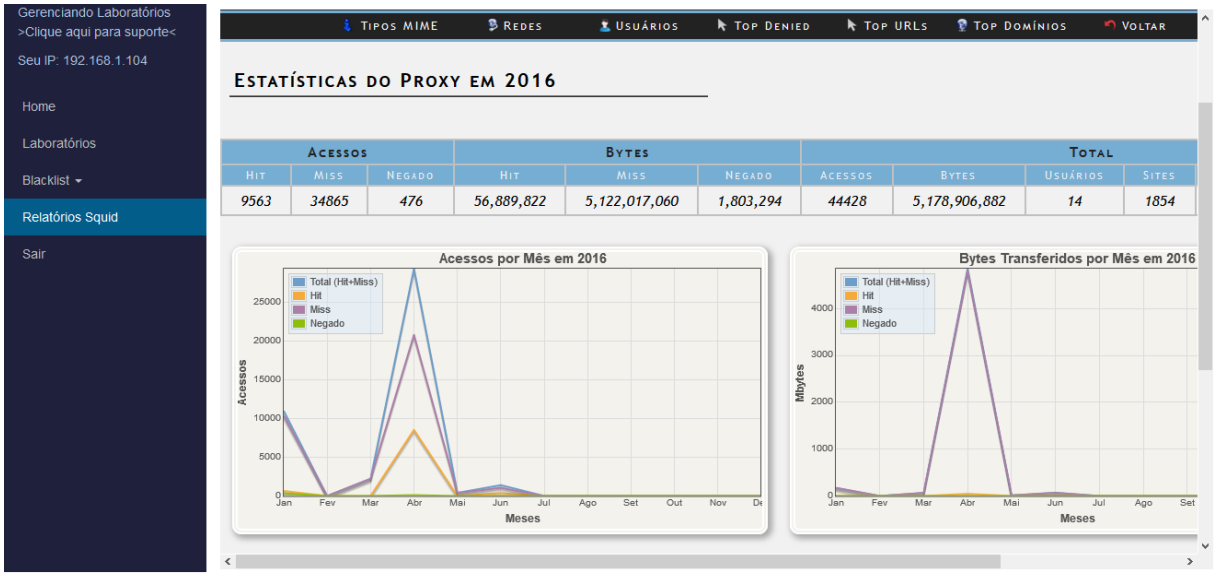

Figura 26 - Relatórios Squid (Squidanalyser).
Fonte: Acervo Pessoal.

5 TESTES E RESULTADOS

Nos testes feitos simulando um laboratório utilizado por três usuários e os mesmos efetuando acessos a internet, através do *proxy*, a ferramenta *web* mostrou-se eficiente na interação com as configurações de grupos pré-determinadas no *Squid*, no bloqueio de *blacklists* para o laboratório selecionado e também o bloqueio dos sites inseridos nos perfis. Para simular um laboratório em uso, foram utilizadas máquinas virtuais e reais, e configurado seus IPs referente a range de ips estabelecida para o laboratório utilizado, no arquivo de configurado do *squidguard*. Através da figura 27 é possível identificar a range de ips estabelecida para cada laboratório e em amarelo o laboratório selecionado para teste (Lab2).

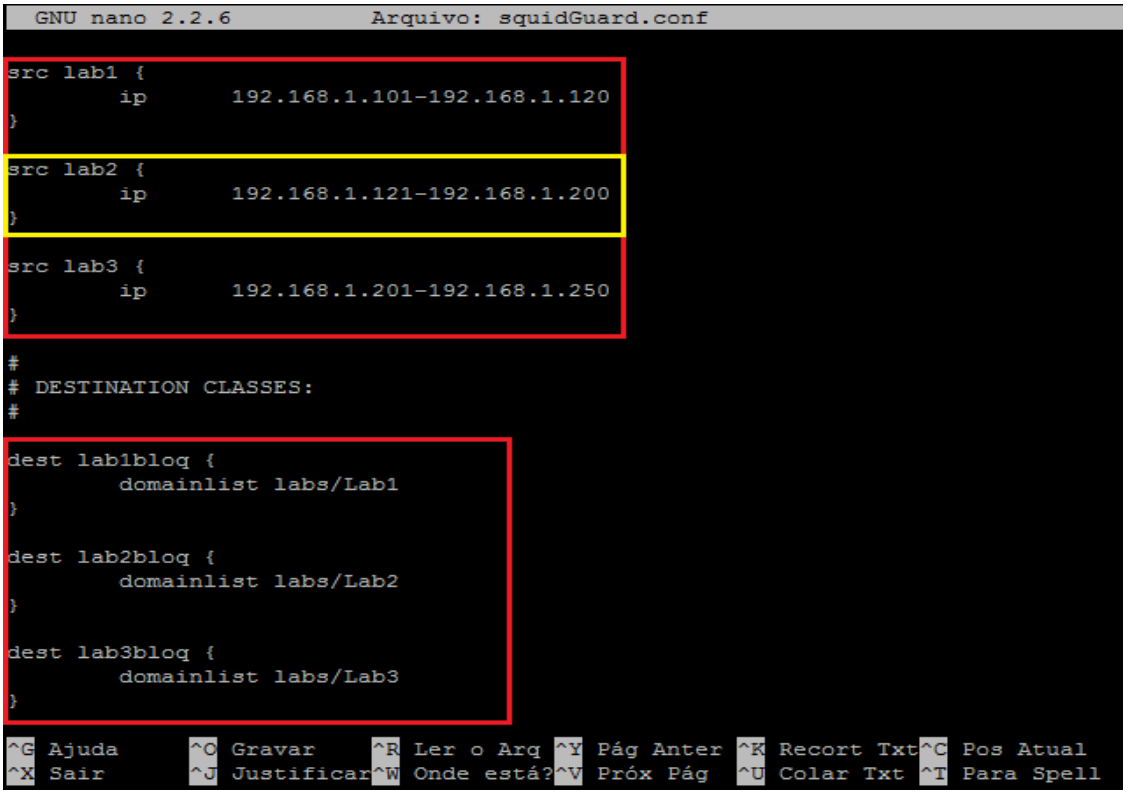

Figura 27 - Range de ips referente a cada laboratório e arquivos que contém os sites bloqueados para tais.
Fonte: Acervo Pessoal.

Já na figura 28, visualiza-se a data e o tempo de aplicação do perfil selecionado para o laboratório.

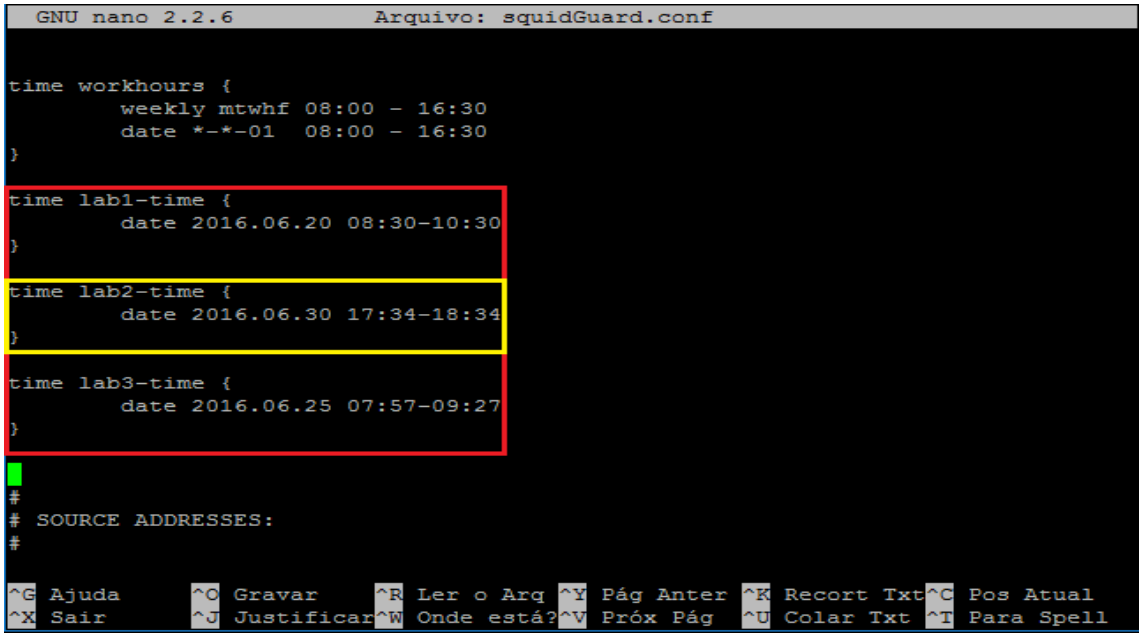

Figura 28 - Dia e horário de aplicação do bloqueio.
Fonte: Acervo Pessoal.

Na figura 29 visualiza-se o horário do sistema no momento em que é aplicado o tempo de bloqueio para o laboratório selecionado. A figura 28 é resultado das informações inseridas na figura 29.

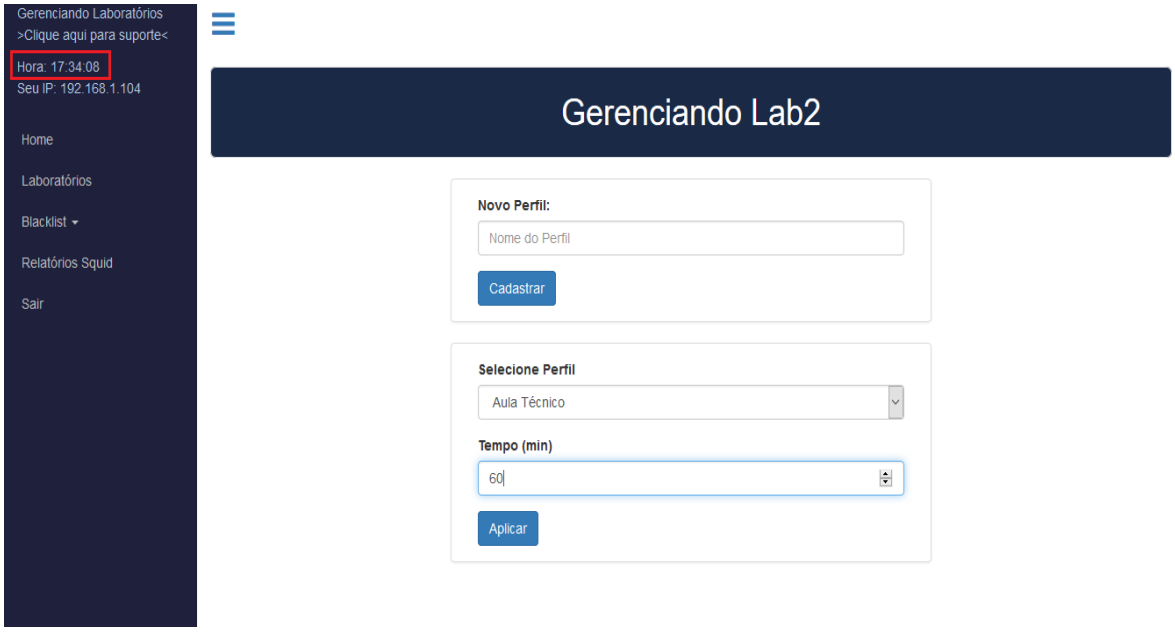

Figura 29 - Aplicando perfil ao Lab2 e verificando seu horário de aplicação.
Fonte: Acervo Pessoal.

Após a aplicação do perfil selecionado ao laboratório utilizado como teste, solicitou-se aos usuários participantes da simulação que acessassem sites de sua escolha. Os registros dos acessos podem ser verificados na figura 30.

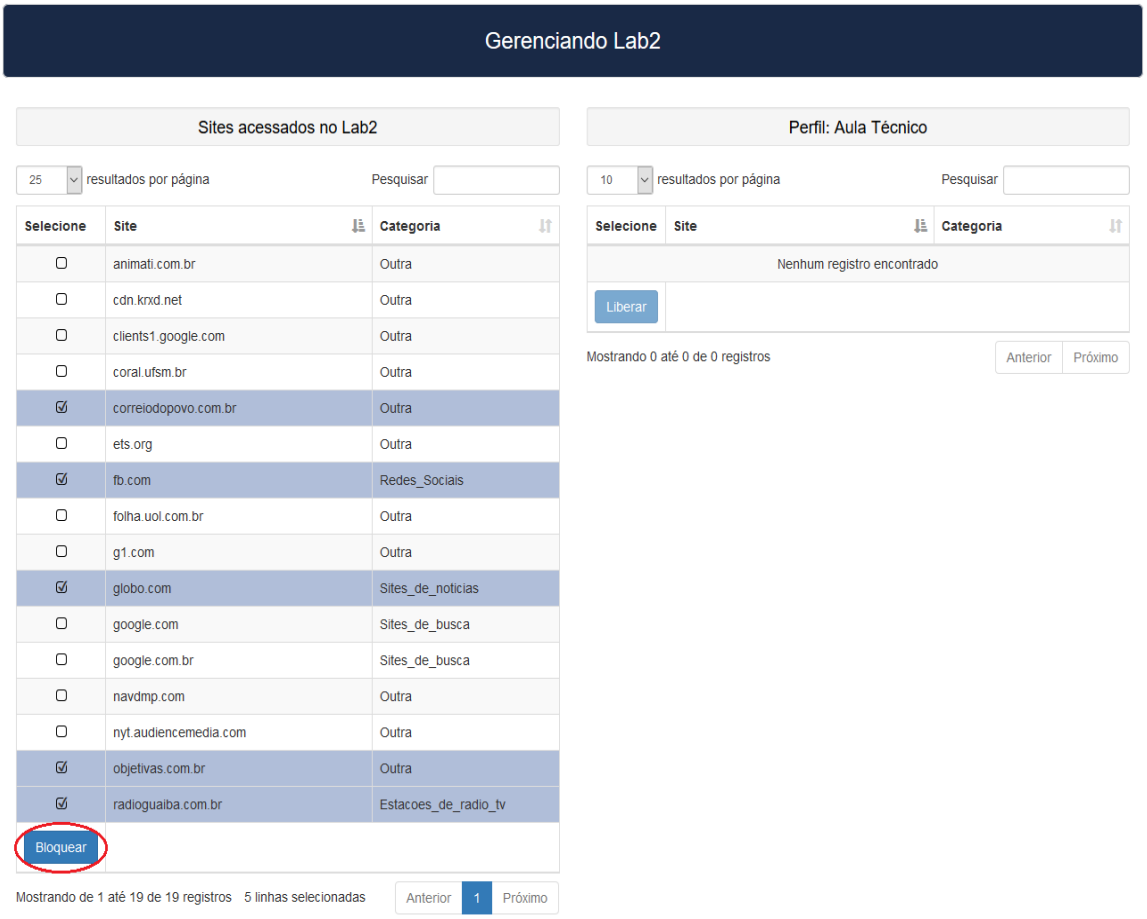

Figura 30 - Sites acessados no Lab2 e selecionados para bloqueio.
Fonte: Acervo Pessoal.

Já na figura 31 é possível visualizar os sites selecionados na figura 30, inseridos no perfil de bloqueio utilizado.

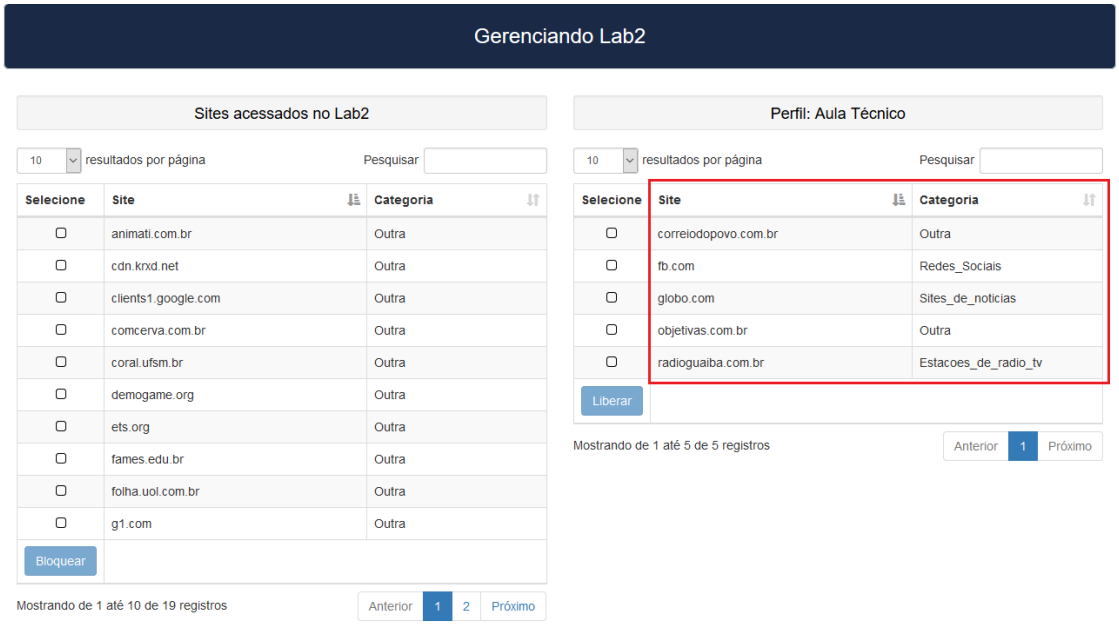

Figura 31 - Visualização do perfil com os sites inseridos para bloqueio.
Fonte: Acervo Pessoal.

Através da figura 32 pode-se verificar a inserção de novos sites para bloqueio.

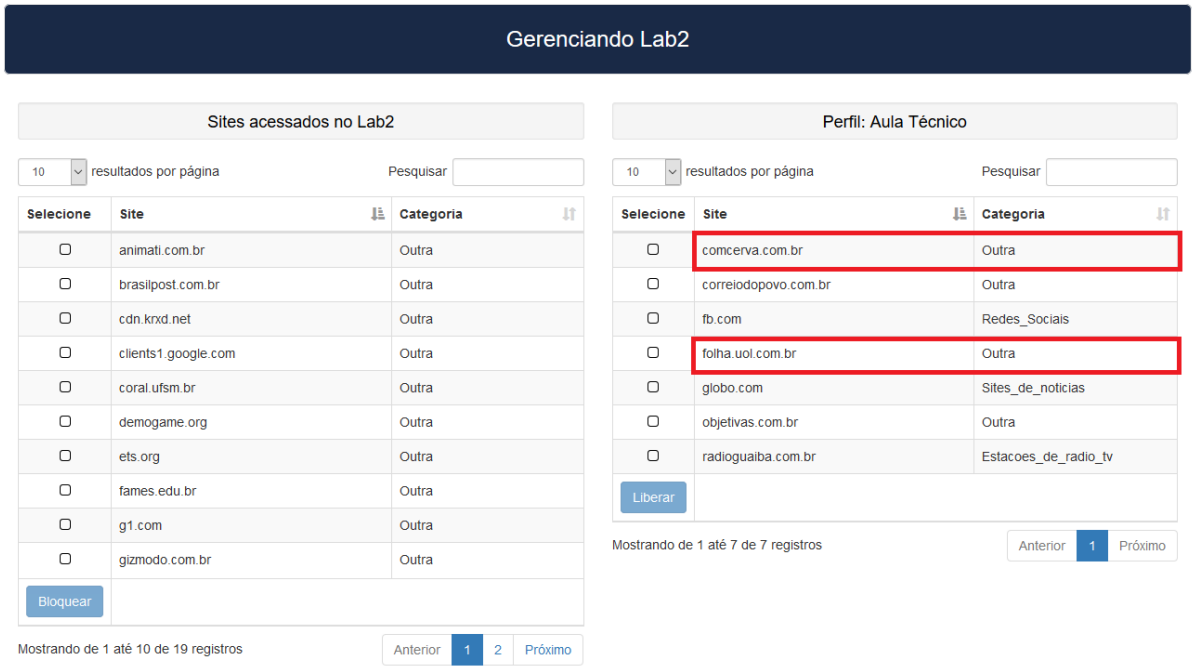

Figura 32 - Inserção de novos sites para bloqueio.
Fonte: Acervo Pessoal.

Finalizado a inserção de sites para bloqueio, solicitou-se para os participantes do teste que acessassem sites inseridos no perfil de bloqueio e outros quaisquer que não estivessem inseridos. As figuras 34, 35 e 36 exemplificam o acesso dos usuários a ambos os tipos de sites e seus possíveis bloqueios. Cada figura representa um usuário diferente e é possível visualizar o horário em que este está efetuando o acesso.

Para melhor compreensão das informações apresentadas nas figuras 34, 35 e 36, mostra-se na figura 33 um quadro contendo o IP de cada usuário, site acessado e horário de acesso.

Usuários	IP	Data - Horário de acesso	Sites acessados	Sites bloqueados	Sites liberados
Usuário 1	192.168.1.122	30/06/2016 - 17:58	comcerva.com.br objetivas.com.br yahoo.com	comcerva.com.br objetivas.com.br	yahoo.com
Usuário 2	192.168.1.125	30/06/2016 - 17:52	fb.com globo.com site.ufsm.br	fb.com globo.com	site.ufsm.br
Usuário 3	192.168.1.127	30/06/2016 - 18:12	correiodopovo.com.br radioguaiba.com.br animati.com.br	correiodopovo.com.br radioguaiba.com.br	animati.com.br

Figura 33 - Informações dos usuários referente as figuras 34, 35 e 36.
Fonte: Acervo Pessoal.

Na figura 34 é possível visualizar as informações referentes ao Usuário 1.

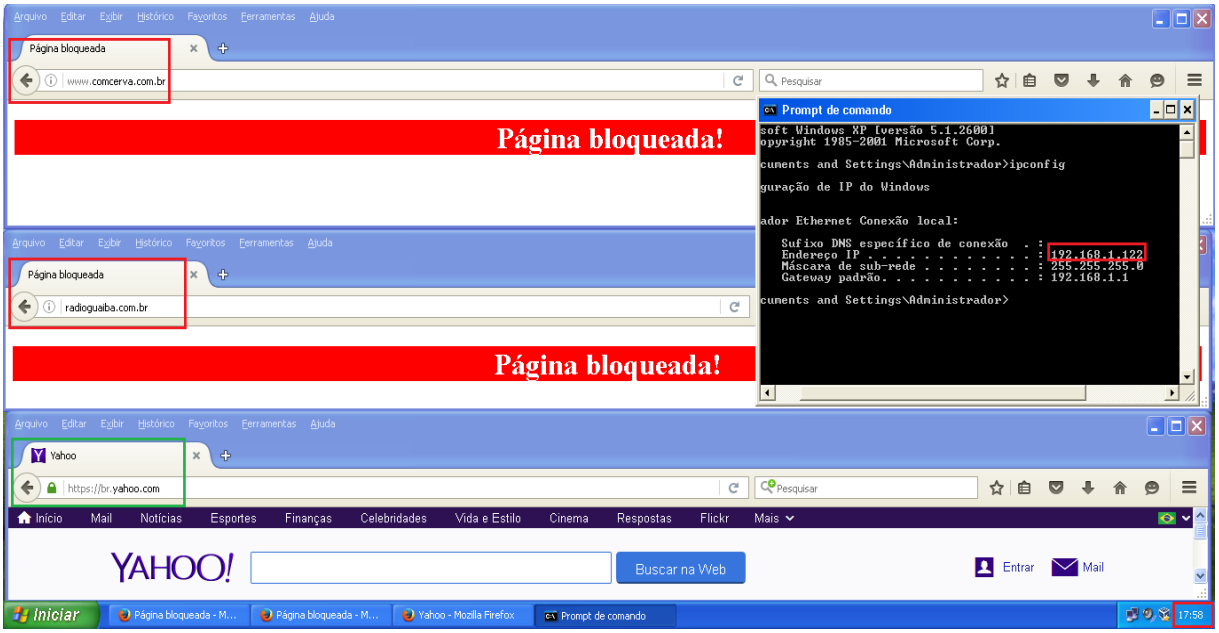

Figura 34 - Acesso do Usuário 1 a duas páginas bloqueadas e uma liberada.
Fonte: Acervo Pessoal.

Já na figura 35 podem ser vistas as informações relacionadas ao Usuário 2.

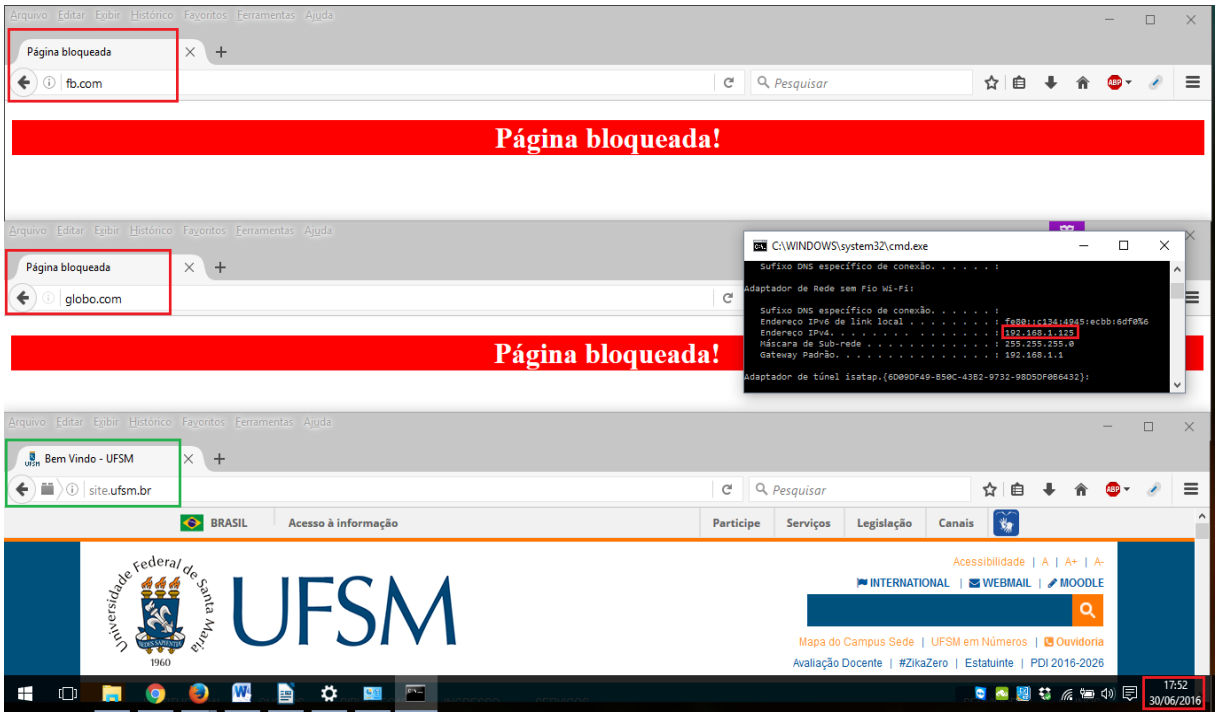

Figura 35 - Acesso do Usuário 2 a duas páginas bloqueadas e uma liberada
Fonte: Acervo Pessoal.

E por fim, na figura 36 visualizam-se as informações referenciadas no Usuário 3.

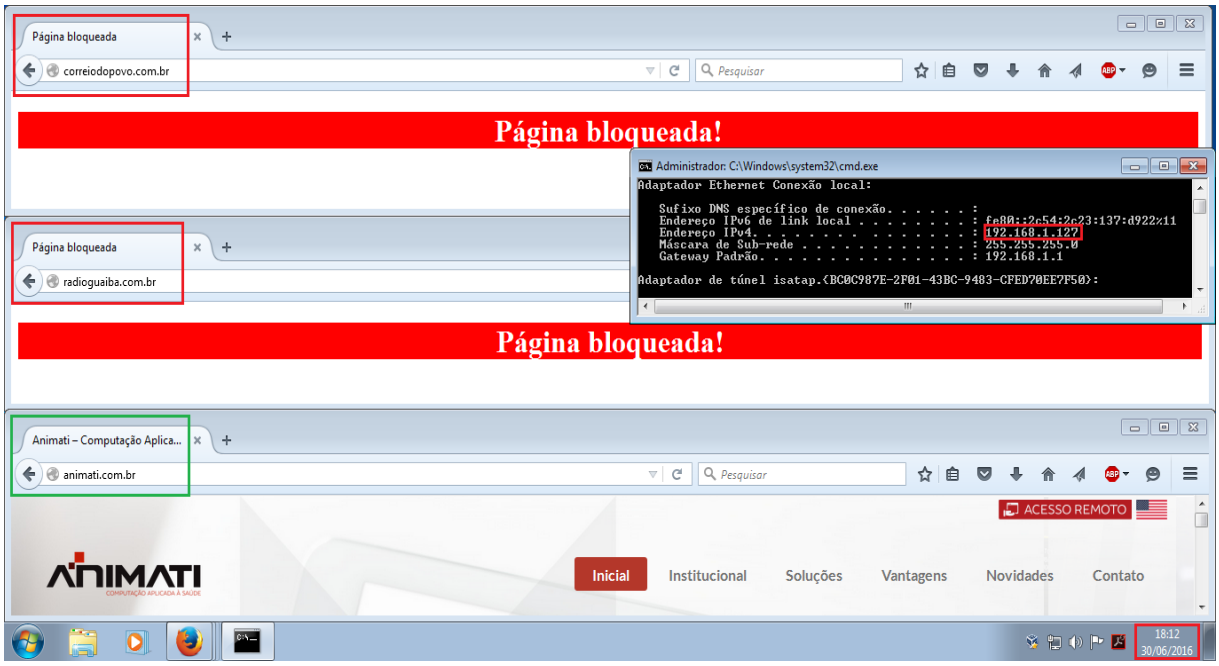

Figura 36 - Acesso do Usuário 3 a duas páginas bloqueadas e uma liberada.
Fonte: Acervo Pessoal.

Uma importante vantagem identificada nessa ferramenta *web* em comparação a ferramenta Carraro Dashboard é a forma como se efetua o bloqueio do acesso. Na Carraro Dashboard é necessário cadastrar os sites, individualmente, nos grupos pré-definidos e inserir usuários que farão parte desse bloqueio, bem como reiniciar o serviço do *Proxy Squid* para que as regras sejam aplicadas. Já na ferramenta apresentada, é listado todo o acesso referente ao laboratório selecionado, podendo ser comparado a um grupo, e o professor terá somente o trabalho de selecionar o site para bloqueio e inserir no seu perfil gerenciável. Em relação ao reinicio do serviço do *Proxy Squid*, é pré-configurado e executado de forma automática em toda ação que envolve a alteração dos seus arquivos de configuração.

Em comparação a ferramenta *SquidGuard Manager*, a ferramenta *web* desenvolvida apresenta uma visão mais privilegiada e menos técnica, visto que, a *SquidGuard Manager* exige uma verificação das regras configuradas e das listas aplicadas as estas, bem como uma reinicialização manual toda vez que forem alteradas, possibilitando assim, que o utilizador cometa um erro ao esquecê-la.

6 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

A proposta apresentada neste trabalho apresentou resultados satisfatórios, considerando que atendeu ao objetivo proposto. Através de uma simulação de monitoramento e controle dos acessos em um laboratório, possibilitou-se que o usuário gerencia-se o acesso ao conteúdo, disponibilizado na internet, durante o período de aula, tendo como forma de controle de acesso a utilização de seus perfis gerenciáveis durante a aplicação da ferramenta. Dessa forma, o usuário conseguiu efetuar um controle de acesso sem possuir grande conhecimento técnico acerca do *software Squid*.

A ferramenta foi desenvolvida levando em consideração o controle de acesso aos sites e a comparação desses com as *blacklists*, de modo a informar para seu utilizador à categoria que esse site se encontra, servindo de auxilio no momento da seleção dos sites que serão inseridos pra bloqueio. Fato que torna a ferramenta *web* distinta das demais já desenvolvidas, já que, as configurações são pré-definidas, sem necessidade de criar usuários ou grupos, manualmente, para serem bloqueados, pois, isso se da através da utilização do perfil gerenciável aplicado aos grupos pré-determinados, que chamou-se de laboratórios no trabalho.

Como proposta para realização de trabalhos futuros, a ferramenta poderá ser implementada em um ambiente real, como laboratórios de aulas, de modo que se obtenham resultados mais satisfatórios. Assim, será possível realizar testes de usabilidade, com o objetivo de atrair ideias para o aperfeiçoamento da ferramenta.

Outra sugestão interessante seria a criação de *whitelists*, que são as listas dos sites que terão o acesso permitido, possibilitando assim, utilizar um processo de bloqueio inverso ao apresentado pela ferramenta desenvolvida, o qual se dá através do bloqueio total dos acessos e a liberação somente do conteúdo que estiver inserido nas *whitelists*.

Para finalizar, uma última consideração pode ser feita: a inserção de um campo informativo que possibilite ao professor visualizar a range de IPs contida no laboratório gerenciado.

7 REFERÊNCIAS BIBLIOGRÁFICAS

ALMEIDA, Maurício. **Uma introdução ao XML, sua utilização na Internet e alguns conceitos complementares,** 2002. Disponível em: <http://www.scielo.br/pdf/ci/v31n2/12903>. Acesso em 25 de maio de 2016.

BAROS, E. B. **Configurando um Squid "ninja".** 2006. Disponível em: <http://www.rjunior.com.br/download/squid-ninja.pdf>. Acesso em 28 de maio de 2016.

MARCOSCARRARO.BLOGSPOT.COM.BR, 2011. Disponível em: < http://marcoscarraro.blogspot.com.br/2011/05/lancado-o-carraro-dashboard.html#.V23gWDWTOPJ>. Acesso em: 10 de junho de 2016.

CONCEIÇÃO, Mateus. **Octopus: ferramenta de código aberto para geração de relatórios para o servidor Proxy Squid,** 2012.

COUTO, André. **Uma abordagem de gerenciamento de redes baseado no monitoramento de fluxos de tráfego netflow com o suporte de técnicas de business intelligence**, 2012. Disponível em: <http://repositorio.unb.br/bitstream/10482/11519/1/2012_AndreValentedoCouto.pdf>. Acesso em 20 de maio de 2016.

DUARTE, D. **Por que proxy não transparente é melhor que o transparente**, 2011 Disponível em: <http://www.purainfo.com.br/hardwareredes/por-que-proxy-notransparente-melhor-que-o-transparente/>. Acesso em 24 de maio de 2016.

DEBIAN.ORG, 2016. Disponível em: <https://www.debian.org/intro/about>. Acessado em: 02 de junho de 2016.

JARGAS, Aurélio. **Introdução ao Shell Script**, 2004. Disponível em: <http://aurelio.net/shell/apostila-introducao-shell.pdf>. Acesso em: 25 de maio de 2016.

KUROSE, J. F.; Ross, K. W. **Redes de Computadores e a Internet:** uma abordagem to*p-d*own. 5. Ed. São Paulo: Addison Wesley, 2010.

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

MCCABE, James. **Network Analysis, Architecture, and Design**. Burlington: Morgan Kauffmann, 2007.

MORIMOTO, Carlos E. **Entendendo e dominando o Linux**.
Guia do Hardware,2009. Disponível em:<http://www.guiadohardware.net>. Acesso em 24 de maio de 2016.

MÜLLER, Bruno. **Sistema Gerenciador de Scripts**, 2011.

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

PETRY. A. C. **Desenvolvimento de firewall com hardware de baixo desempenho e fácil configuração em nível de usuário**, 2013.

SIEWERT, V. C. **Ferramenta web para administração do servidor proxy squid**, 2007.

SILVA, M. S. **JavaScript**: Guia do Programador. São Paulo: Novatec Editora, 2010.

SquidAnalyzer.darold.net, 2014. Disponível em: <http://squidanalyzer.darold.net>. Acessado em: 02 de junho de 2016.

SquidGuard.org. 2016. Disponível em: <http://www.squidguard.org/>. Acessado em: 02 de junho de 2016.

STALLINGS, William. **Criptografia e segurança de redes**. Ed. Pearson Prentice Hall, 2008.

TANENBAUM, A. S. **Redes de Computadores** 4. Ed. Rio de Janeiro: Elsevier, 2003.

THOMAS, T. **Segurança de Redes:** Primeiros passos. Rio de Janeiro: Editora Ciência Moderna Ltda, 2007.

WATANABE, C. S. **Introdução ao cache de web**. Rio de Janeiro, 2000. Disponível em: <https://memoria.rnp.br/newsgen/0003/cache.html>. Acesso em 23 de maio de 2016.

WESSELS, Duane. **Squid: The Definitive Guide**. O'Reilly, 2004.

WOLF, L. A.; SILVA, V. C. O. da. **Controle de acesso em segmentos de rede para usuários autorizados em um ambiente corporativo**, Universidade Luterana do Brasil (ULBRA), Curso Superior de Tecnologia Em Redes De Computadores – Campus Canoas, Junho de 2011.

8 APÊNDICE

Apêndice A – Acessar internet via *Proxy* pelo aluno.

Descrição	Acesso *web*.
Ator	Usuário.
Pré-condição	Iniciar acesso.
Fluxo principal	a) Abrir navegador; b) Configurar ip do *proxy* no navegador; c) Efetuar acesso desejado.
Fluxo Alternativo	a) Local acessado é inválido; b) Servidor *proxy* configurado incorretamente; - Retornar ao passo "b" do fluxo principal. c) Local acessado está bloqueado.
Pós-condição	O usuário acessa o local desejado.

Apêndice B - Acessar ferramenta *web* pelo professor.

Descrição	Acessar ferramenta *web*.
Ator	Professor.
Pré-condição	Iniciar acesso.
Fluxo principal	a) Abrir navegador; b) Informar endereço IP da ferramenta *web*; c) Inserir credenciais; - Login ou senha inválidos; d) Iniciar acesso; e) Utilizar menus para navegar.
Fluxo Alternativo	a) Login ou senha inválidos; - Verificar dados e retornar ao passo "b" do fluxo principal. - Efetuar cadastro na página de login.
Pós-condição	Professor acessa a ferramenta *web*.

Apêndice C – Selecionar o laboratório para gerenciar

Descrição	Acessar página dos laboratórios.
Ator	Professor.
Pré-condição	Iniciar acesso e informar as credenciais; Menu “Laboratórios”; Selecionar laboratório.
Fluxo principal	a) Abrir navegador; b) Inserir credenciais; - Login ou senha inválidos; c) Iniciar acesso; d) Acessar o menu “Laboratórios”; e) Selecionar laboratório.
Fluxo Alternativo	a) Login ou senha inválidos; - Verificar dados e retornar ao passo “b” do fluxo principal. - Efetuar cadastro na página de login.
Pós-condição	Professor acessa a página dos laboratórios.

Apêndice D – Selecionar perfil

Descrição	Acessar página de seleção do perfil.
Ator	Professor.
Pré-condição	Iniciar acesso e informar as credenciais; Menu “Laboratórios”; Selecionar laboratório.
Fluxo principal	a) Abrir navegador; b) Inserir credenciais; - Login ou senha inválidos; c) Iniciar acesso; d) Acessar o menu “Laboratórios”; e) Selecionar laboratório; f) Cadastrar perfil; g) Selecionar perfil; - Nenhum perfil cadastrado; h) Inserir tempo de bloqueio; i) Aplicar perfil selecionado ao laboratório.
Fluxo Alternativo	a) Login ou senha inválidos; - Verificar dados e retornar ao passo “b” do fluxo principal. - Efetuar cadastro na página de login. b) Nenhum perfil cadastrado; - Retornar ao passo “f” do fluxo principal.
Pós-condição	Professor aplica o perfil selecionado ao laboratório.

Apêndice E – Bloquear sites do laboratório selecionado

Descrição	Acessar página de gerenciamento do laboratório.
Ator	Professor.
Pré-condição	Iniciar acesso e informar as credenciais; Menu “Laboratórios”; Selecionar laboratório.
Fluxo principal	a) Abrir navegador; b) Inserir credenciais; - Login ou senha inválidos; c) Iniciar acesso; d) Acessar o menu “Laboratórios”; e) Selecionar laboratório; f) Cadastrar perfil; g) Selecionar perfil; - Nenhum perfil cadastrado; h) Inserir tempo de bloqueio; i) Aplicar perfil selecionado ao laboratório; j) Selecionar sites para bloqueio; k) Clicar em bloquear;
Fluxo Alternativo	a) Login ou senha inválidos; - Verificar dados e retornar ao passo “b” do fluxo principal. - Efetuar cadastro na página de login. b) Nenhum perfil cadastrado; - Retornar ao passo “f” do fluxo principal.
Pós-condição	Professor bloqueia os sites para o laboratório e insere no perfil selecionado.

Apêndice F – Bloquear Blacklist

Descrição	Bloquear Blacklist para o laboratório selecionado.
Ator	Professor.
Pré-condição	Iniciar acesso e informar as credenciais; Menu “Laboratórios”; Selecionar laboratório; Aplicar perfil; Menu “Blacklist”; Opção “Bloquear”; Selecionar Blacklist(s); Bloquear.
Fluxo principal	a) Abrir navegador; b) Inserir credenciais; - Login ou senha inválidos; c) Iniciar acesso; d) Acessar o menu “Laboratórios”; e) Selecionar laboratório; f) Cadastrar perfil; g) Selecionar perfil; - Nenhum perfil cadastrado; h) Inserir tempo de bloqueio; i) Aplicar perfil selecionado ao laboratório; j) Acessar o menu “Blacklist”; k) Selecionar opção “Bloquear”; - É possível visualizar todas as *blacklists* liberadas para o laboratório selecionado. l) Selecionar *blacklists*; m) Clicar em bloquear;
Fluxo Alternativo	a) Login ou senha inválidos; - Verificar dados e retornar ao passo “b” do fluxo principal. - Efetuar cadastro na página de login. b) Nenhum perfil cadastrado; - Retornar ao passo “f” do fluxo principal. c) Nenhuma *blacklist* é listada;

	- Retornar ao passo “e” do fluxo principal.
Pós-condição	Professor bloqueia as *blacklists* para o laboratório selecionado.

Apêndice G – Liberar *Blacklist*

Descrição	Liberar *Blacklist* para o laboratório selecionado.
Ator	Professor.
Pré-condição	Iniciar acesso e informar as credenciais; Menu “Laboratórios”; Selecionar laboratório; Aplicar perfil; Menu “Blacklist”; Opção “Liberar”; Selecionar Blacklist(s); Liberar.
Fluxo principal	a) Abrir navegador; b) Inserir credenciais; - Login ou senha inválidos; c) Iniciar acesso; d) Acessar o menu “Laboratórios”; e) Selecionar laboratório; f) Cadastrar perfil; g) Selecionar perfil; - Nenhum perfil cadastrado; h) Inserir tempo de bloqueio; i) Aplicar perfil selecionado ao laboratório; j) Acessar o menu “Blacklist”; k) Selecionar opção “Liberar”; - É possível visualizar todas as *blacklists* bloqueadas para o laboratório selecionado. l) Selecionar *blacklists*; m) Clicar em liberar;
Fluxo Alternativo	a) Login ou senha inválidos;

	- Verificar dados e retornar ao passo “b” do fluxo principal. - Efetuar cadastro na página de login. b) Nenhum perfil cadastrado; - Retornar ao passo “f” do fluxo principal. c) Nenhuma *blacklist* é listada; - Retornar ao passo “e” do fluxo principal.
Pós-condição	Professor bloqueia as *blacklists* para o laboratório selecionado.

Apêndice H – Gerar relatórios

Descrição	Gerar relatórios Squid.
Ator	Professor.
Pré-condição	Iniciar acesso e informar as credenciais; Menu “Relatórios Squid”;
Fluxo principal	a) Abrir navegador; b) Inserir credenciais; - Login ou senha inválidos; c) Iniciar acesso; d) Acessar o menu “Relatórios Squid”;
Fluxo Alternativo	a) Login ou senha inválidos; - Verificar dados e retornar ao passo “b” do fluxo principal. - Efetuar cadastro na página de login.
Pós-condição	Professor efetua geração de relatórios de acesso do *Proxy* Squid.

Apêndice I - Shell script utilizado para geração das tabelas de acessos dos laboratórios.

```
#!/bin/bash


#padrão de log criado para visualizar os dados legivelmente
log="/var/log/squid3/teste.log"
DIR_BLIST="/var/lib/squidguard/db/"
#lista de blacklists e suas traduções
BLDETAILS="/etc/squid3/blacklists-details.txt"
all_sites_lab="/var/www/tcc/arquivos/all"
```

```
perfil_lab="/var/www/tcc/arquivos/labs/"
lab_squidguard="/var/lib/squidguard/db/labs/"
#script que reinicializa o squid e squidguard
squid_rec="/usr/sbin/squid_rec"

#arquivos /var/www/tcc/arquivos/labs/Lab*.txt
#São os perfis carregados.
#Quando varrer o log do squid deverá verificar se o site está inserido nesses arquivos.
#Se estiver, não será adicionado.
#Se não estiver, adiciona no allLab*.
#Logo após, copia o conteúdo desse arquivo para o de bloqueio do squidguard (abaixo).
#Esse arquivo que irá determinar os sites a serem bloqueados no laboratório.

#arquivos /var/www/tcc/arquivos/Lab*time.txt
#Contém o tempo que será aplicado o perfil selecionado
#Após esse tempo deverá ser zerado o arquivo abaixo:
#Arquivo que contém os sites inseridos no perfil para bloqueio no squidguard.
#/var/lib/squidguard/db/labs/Lab*

date=`date +%Y-%m-%d`

alllabs_time=`ls -lh --full-time $all_sites_lab*`

   ORI_IFS=$IFS
        IFS=$'\n'
        for line in $alllabs_time; do
          alldate=`echo $line | awk '{print $6}'`
          allname=`echo $line | awk '{print $9}'`

          if [ "$alldate" != "$date" ]
            then
            echo -e "{\n\"data\": [\n]\n}" > $allname
          fi
```

```
        done
   IFS=$ORI_IFS

timenow=`date +%H:%M:%S`
horaagora=`echo $timenow | cut -d: -f1`

 if [[ $horaagora -ge 7 && $horaagora -le 12 ]]
  then
    timebegin="07:00:00"
    timeend="12:59:59"
 else
   if [[ $horaagora -ge 13 && $horaagora -le 19 ]]
    then
      timebegin="13:00:00"
      timeend="18:59:59"
   else
     timebegin="19:00:00"
     timeend="23:59:59"
   fi
 fi

#esta sendo verificado o turno que o site foi acessado
#07:00:00 - 12:59:59 = manha
#13:00:00 - 18:59:59 = tarde
#19:00:00 - 23:59:59 = noite

ip_user=`awk -v data="$date" -v timenow="$timebegin" -v timeafter="$timeend" '{if
(((($14) == "TCP_MISS/200" || ($14) == "TCP_MISS/301" || ($14) == "TCP_MISS/302")
&& ($5) == data && ($17) == "text/html" && (($8) >= timenow && ($8) <= timeafter ))
print $2}' $log`
site_user=`awk -v data="$date" -v timenow="$timebegin" -v timeafter="$timeend" '{if
(((($14) == "TCP_MISS/200" || ($14) == "TCP_MISS/301" || ($14) == "TCP_MISS/302")
&& ($5) == data && ($17) == "text/html" && (($8) >= timenow && ($8) <= timeafter ))
```

```
print $11}' $log | sed 's/.*:\/\/// ; s/\/.*//' | sed 's/.*www.//'`
ip_site=`paste <(echo "$ip_user") <(echo "$site_user") | sort -f | uniq`

if [ -n "$ip_user" ]
 then

   ORI_IFS=$IFS
     IFS=$'\n'
     for line in $ip_site; do
          user=`echo $line | awk '{print $1}'`
          domain=`echo $line | awk '{print $2}'`

        #verifica em qual categoria o site se encaixa
        #se não estiver em nenhuma categoria das blacklists utilizadas
        #será atribuída a categoria "Outra"
        search_site=`grep -rx "$domain" --exclude="Lab*" --exclude="COPYRIGHT" --
exclude="global_usage" "$DIR_BLIST" | head -1 | cut -d'/' -f6`
          if [ "$search_site" == "" ]
             then
               categ_site=Outra
          else
            if [[ "$search_site" == "automobile" || "$search_site" == "education" ||
"$search_site" == "finance" || "$search_site" == "hobby" || "$search_site" == "recreation"
|| "$search_site" == "science" || "$search_site" == "sex" ]]
             then
             search_site=`grep -rx "$domain" --exclude="Lab*" --exclude="COPYRIGHT" -
-exclude="global_usage" "$DIR_BLIST" | head -1 | cut -d'/' -f7`
            fi
             categ_site=`grep "$search_site" "$BLDETAILS" | awk '{print $2}'`
          fi

        #verifica em qual range de IPs o usuário se encontra
        #para inserir o site no arquivo allLab* correto
```

```
        name_lab=`awk 'BEGIN {
            if (("""$user""") >= "192.168.1.100" && ("""$user""") <= "192.168.1.120") print
("Lab1");
            else if (("""$user""") >= "192.168.1.121" && ("""$user""") <= "192.168.1.200")
print ("Lab2");
            else if (("""$user""") >= "192.168.1.201" && ("""$user""") <= "192.168.1.250")
print ("Lab3");
            }`

            #verifica se o site esta no perfil selecionado para o Lab*
            #no arquivo /var/www/tcc/arquivos/labs/Lab*.txt
            #se não estiver ele verifica se já consta no arquivo allLab*
             grep -x "$domain" "$perfil_lab$name_lab.txt" >> /dev/null
             if [ $? -eq 1 ]
               then
                 #site não esta no arquivo de perfil que esta sendo utilizado
              grep -x "\"$domain\"," "$all_sites_lab$name_lab" >> /dev/null
              if [ $? -eq 1 ]
                then
                  #site não esta no arquivo allLab*
                  #será adicionado ao arquivo allLab*
                  sed -i '3i\[\n"'$domain'",\n"'$categ_site'"\n],' "$all_sites_lab$name_lab"
              fi
             fi
    done
    IFS=$ORI_IFS
fi

   #verifica se tem virgula na antepenúltima linha do arquivo datatable e remove
   #esse teste é feito devido a garantia do padrão datatable utilizado na criação da tabela
   #se algo estiver fora do padrão, o PHP irá informar um aviso de erro na interface web
   allLabs=`ls $all_sites_lab*`
   ORI_IFS=$IFS
```

```
    IFS=$'\n'
    for line in $allLabs; do

      virg=`cat -n "$line" | tail -n3 | head -n1 | grep -w "," | awk '{print $1}'`
      if [ "$virg" != "" ]
       then
         sed -i ""$virg"s/\,//" "$line"
      fi
    done
IFS=$ORI_IFS

#copia o conteúdo dos sites inseridos no perfil
#para o arquivo de bloqueio do squidguard
qtdperfil=`ls $perfil_lab | grep "Lab[0-9]\{,\}.txt" | wc -l`
for i in $(seq $qtdperfil); do
  cat "$perfil_lab"Lab"$i".txt > "$lab_squidguard"Lab"$i"
done

#executa script que reinicializa o squid e squidguard
$squid_rec 2> /dev/null
```