CENTRO UNIVERSITÁRIO UNIFACVEST
CURSO DE CIÊNCIA DA COMPUTAÇÃO
RODRIGO RIBEIRO

FIREWALL IPTABLES E SQUID WEB PROXY SHELL APLICADO AO UBUNTU LINUX: COLÉGIO MARIA IMACULADA

Lages- SC
2016

guigo_ribeiro@hotmail.com

RODRIGO RIBEIRO

FIREWALL IPTABLES E SQUID WEB PROXY SHELL APLICADO AO DEBIAN LINUX: COLÉGIO MARIA IMACULADA

Projeto apresentado à Banca Examinadora do Trabalho de Conclusão de Curso II de Ciência da Computação para análise e aprovação.

Lages- SC
2016

RODRIGO RIBEIRO

FIREWALL IPTABLES E SQUID WEB PROXY SHELL APLICADO AO DEBIAN LINUX: COLÉGIO MARIA IMACULADA

Trabalho de Conclusão de Curso de Ciência da Computação apresentado ao Centro Universitário UNIFACVEST como parte dos requisitos para obtenção do título de bacharel em Ciência da Computação.

Orientador: Prof. Me Márcio José Sembay
Co-Orientador: Prof. Cassandro Albino Devenz

Lages, SC ___/___/2016.

Nota_____

_____

Coordenador do curso de graduação

Lages -SC

2016

AGRADECIMENTOS

Agradeço, à Deus por ter me dado saúde e força para superar dificuldades, a minha família: Neusa Ribeiro, Manoel Pedro Ribeiro e Marcia Daniela e a minha namorada Carolina Melo Menegotto, pelo apoio e incentivo em todos os momentos para tornar meus objetivos possíveis

Aos meus orientadores pelo suporte no pouco tempo que lhe coube, pelas suas correções e incentivos criando um ambiente favorável para o desenvolvimento deste projeto.

Aos meus colegas de aula, pelo companheirismo demonstrado ao passar destes anos. Espero que todos façamos algo para contribuir com o avanço do mundo da Computação.

RESUMO

Este trabalho tem como finalidade a investigação sobre técnicas de desenvolvimento de ferramentas em plataforma de código aberto, será apresentado duas ferramentas em torno da área da administração de redes de computadores em busca de avaliar a segurança de sistemas computacionais e os problemas que a falta de segurança em uma corporação pode ocasionar, para tais propósitos será demonstrado como ferramentas nativas da plataforma Linux são de extrema importância e as vantagens da plataforma de código aberto pode proporcionar. A partir de um conjunto de instruções aplicado ao Debian Linux, desenvolver regras e ferramentas de firewall, pois através delas é possível identificar falhas de segurança e consequentemente corrigi-las. O objetivo do mecanismo de segurança baseado no conjunto destes softwares é proporcionar maior confiabilidade, segurança dos dados trafegados, melhorar desempenho da conexão, proteção contra intrusão, agilidade, proporcionada pelo servidor Proxy e filtro de acesso a conteúdo. O objeto de estudo visa amenizar a complexidade, mostrando que é possível a interação entre usuário e sistema através de comandos com técnicas diferenciadas.

Palavras Chave: *Firewall. Linux.* Redes de Computadores.

ABSTRACT

This work aims to research tools open source platform development techniques will be presented two tools around the area of administration of computer networks for evaluating the security of computer systems and the problems that the lack of security in a corporation may result, for such purposes will be demonstrated as native tools on the Linux platform are of utmost importance and the advantages of the open source platform can provide. From a set of instructions applied to Debian Linux firewall rules and develop tools because it is possible through them to identify security flaws, and consequently to correct them. The purpose of the security mechanism based on the set of this software is to provide greater reliability, security of transmitted data, improve performance of the connection, intrusion protection, agility, provided by the proxy server and filter access to content. The object of study aims to alleviate the complexity, showing the interaction between user and system commands via different techniques are possible.

Key Words: *Firewall. Linux. Computer Network.*

RESUMEN

Este trabajo tiene como objetivo herramientas de investigación de técnicas de desarrollo de la plataforma de código abierto se presentarán dos herramientas en todo el ámbito de la administración de las redes informáticas para evaluar la seguridad de los sistemas informáticos y los problemas que la falta de seguridad en una corporación puede resultar, para tales fines se demostrará como herramientas nativas de la plataforma Linux son de suma importancia y las ventajas de la plataforma de código abierto pueden proporcionar. A partir de un conjunto de instrucciones aplicadas a las reglas del cortafuegos para Debian Linux y desarrollar herramientas ya que es posible a través de ellos para identificar las fallas de seguridad, y en consecuencia para corregirlos. El propósito del mecanismo de seguridad basado en el conjunto de este programa es proporcionar una mayor fiabilidad, la seguridad de los datos transmitidos, mejorar el rendimiento de la conexión, protección contra intrusos, agilidad, proporcionada por el servidor proxy y el acceso a los filtros de contenido. El objeto de estudio tiene como objetivo aliviar la complejidad, que muestra la interacción entre los comandos de usuario y de sistema a través de diferentes técnicas son posibles.

Palabras Clave: *Cortafuegos. Linux. Redes de Computadores.*

LISTA DE FIGURAS

LISTA DE SIGLAS

ACL	-	*Access Control List*
ARP	-	*Address Resolution Protocol*
DHCP	-	*Dynamic Host Configuration Protocol*
DNS	-	*Domain Name System*
FTP	-	*File Transfer Protocol*
GB	-	*Giga Byte*
HTTP	-	*HyperText Transfer Protocol*
ICMP	-	*Internet Control Message Protocol*
IP	-	*Internet Protocol*
IPSEC	-	*IP Security Protocol*
LAN	-	*Local Área Network*
NAT	-	*Network Address Translation*
OSI	-	*Open Systems Interconnection*
PHP	-	*Personal Home Page ou Hypertext Preprocessor*
SH	-	*Shell Script*
SSH	-	*Security Shell*
SSL	-	*Socket Socket Layer*
TCP	-	*Transmission Control Protocol*
UDP	-	*User Datagram Protocol*
URL	-	*Uniform Resource Locator*
WWW	-	*World Wide Web*

SUMÁRIO

1 INTRODUÇÃO

A internet surgiu no final da década de 60 durante a guerra fria com objetivo estritamente militares, ela foi criada para interligar bases militares dos Estados Unidos e com isso garantir que as comunicações norte americanas seriam mantidas mesmo em caso de ataques inimigos que destruíssem os meios convencionais de telecomunicações (TANENBAUM, 2003).

O Colégio Maria Imaculada é uma das milhares instituições de ensino que trabalha diretamente por meio da web, atualmente desta podemos perceber que o risco fica exposto a partir do momento em que se inicia uma conexão, as diversas análises que foram feitas no decorrer deste trabalho foi em evidência dos fatos em que mesma só utiliza equipamentos para fazer a comunicação entre os computadores e não possui uma política de segurança, com isso a inexistência de equipamentos de segurança torna o colégio insuficientemente capaz de prevenir invasões, ataques mais sofisticados ou qualquer outro tipo de software malicioso, colocando assim à integridade dos dados do colégio em total risco.

Para justificar esse fato, atualmente é fundamental para uma empresa realizar acessos por meio da web e por essa razão a internet teve popularização e um crescimento exponencial, por conta disso essa questão passou a ter várias mudanças no cenário atual, foram criadas ferramentas que assegurassem os dados dos usuários frente às ameaças existentes na rede mundial de computadores motivando os administradores de sistemas a se preocupar com a segurança de seus servidores.

O termo segurança é frequentemente usado com o significado de anular ou minimizar as chances de um invasor obter acesso a um sistema, a segurança está relacionada à proteção contra acesso ou manipulação, intencional ou não, de informações por elementos não autorizados, ou seja, invasores (STALLINGS, 2008).

Dentro deste contexto implantação de servidores ou implementação de uma política de segurança interna, minimiza ou reduz de forma significativa a questão de segurança desde que se desenvolva bons softwares de gerenciamento e proteção da rede.

1.1 Justificativa

O motivo pelo qual esse tema foi escolhido foi para reforçar a ideia de que a administração de redes e a segurança de sistemas em uma corporação é de extrema relevância, podemos afirmar que segurança computacional é parte dos negócios de qualquer empresa de pequeno, médio ou grande porte, principalmente quando falamos de segurança de dados que exigem sigilo.

É nesse propósito que será dado soluções significativas para se ter um sistema com a segurança minimizada e corrigido de falhas, entretanto será apresentada as inúmeras vantagens que a plataforma Linux tem sobre outros sistemas quando o quesito segurança de sistemas se traz à tona, levando em conta sua robustez por natureza e suas distribuições que é escolha de milhões de servidores do mundo inteiro.

Este trabalho será desenvolvido buscando facilitar e otimizar tarefas rotineiras que o administrador de rede tem através da autenticação dos usuários que fazem parte servidor, portanto podemos afirmar que interface gráfica para administrador de rede ainda é um tabu a ser quebrado, pois todo administrador de rede é de modo um utilizador de códigos via linha de comando como forma nativa do Linux.

Ainda nessa necessidade será criado uma interface background na linguagem orientado a objetos *PHP* (SQUID WEB PROXY SHELL) que é um facilitador do servidor de proxy Squid que atualmente e gerenciado via linha de comando no Shell do Linux, atualmente rodando na versão estável *3.5 STABLES*, procurando aumentar o nível de proteção do servidor uma implementação de script de Firewall (FIREWALL/IPTABLES) pensando na mesma ideia de proteção, de forma a incentivar os utilizadores de software livre a alterar, modificar, customizar e aprimorar conforme sua necessidade, como forma de incentivo aos recurso deste poderoso sistema operacional de código aberto.

Vemos que a internet não é um lugar muito seguro, nela, os delinquentes estão por toda parte, criando todo tipo de destruição. Dada a natureza hostil da internet, vamos considerar uma rede de organização e um administrador de rede que a administra (KUROSE, 2010).

1.2 Importância

Pela liberdade, robustez, alto desempenho que o sistema proporciona e estabilidade quando se pretende alcançar qualidade dos dados, pode-se concluir que fornecer técnicas importantes de segurança baseadas em firewall dentro de um sistema operacional Linux, como soluções a serem priorizadas no âmbito da tecnologia de redes e nos vários estados da internet, com o propósito de se obter uma boa segurança das informações a um custo baixo e desempenho superior a outros sistemas.

O grande propósito da ferramenta que irei criar não é somente privar o usuário do uso geral de ferramentas que a internet possibilita e sim proteger os bens em um cenário privado como é o das empresas nos dias de hoje, nenhum um sistema está livre de intrusões, parada de servidores sem backup, ainda mais quando se tem dados importantes trafegando pela rede mundial, por meio disso o projeto irá facilitar tarefas rotineiras do administrador de rede, assim como proteger os bens de uma corporação, vale ressaltar que além de você navegar e trabalhar normalmente no dia-dia, é de extrema importância que você tenha confiabilidade e qualidade de conexão (MARIMOTO, 2004).

1.3 Objetivos Gerais

O objetivo geral deste trabalho é o desenvolvimento de duas ferramentas em ambiente Linux baseado na implantação de um servidor de segurança no Colégio Maria Imaculada, visando proteção e gerenciamento dos processos de rede. O objetivo inicial é proteger a rede com firewall e ao mesmo tempo transformar certas ferramentas em um facilitador para os usuários realizar processos seguros na rede.

1.3.1 Objetivos Específicos

O objetivo específico da pesquisa é apresentar uma visão ampla sobre a importância da segurança e como implantá-la, mostrando os fundamentos de busca por segurança da informação em empresas na atualidade e como isso muda totalmente o modelo de visão e o andamento dos negócios, é possível perceber que a cada momento cresce o número de ameaças que transitam pela rede e para que a qualidade na segurança se torne realidade é necessário reforçar o modelo de segurança. Objetivo deste trabalho é

explorar o máximo que a ferramenta de código livre pode oferecer e nesse sentido evitar e detectar possíveis anomalias que possam transitar no tráfego da rede criando regras específicas de um eficaz modelo de segurança computacional.

O conteúdo listado abaixo contém as principais características e funcionalidades que serão desenvolvidos no decorrer deste projeto:

a) **Firewall com IPTABLES/NETFILTER:** Examinar o tráfego enquanto ele entra em umas das suas interfaces de rede e aplicar regras ao tráfego, essencialmente, permitindo ou impedindo o tráfego baseado nestas regras.
b) **Squid com interface visual:** Um painel de administração para o servidor de rede, que dá ao administrador uma opção rápida e unificada para monitorar e manter regras de Squid, Reduzir a utilização da conexão e melhorar o tempo de resposta a páginas web.
c) **Monitoramento através de NTOP e Wireshark:** Realizar captura e análise de pacotes de rede detecção de intrusão e acompanhamento dos usuários do Colégio gerando logs de relatórios e estatísticas da rede. Gerar resultado na tela do servidor uma filtragem em tempo real de todos os pacotes que estão trafegando pela rede.

O tópico a seguir será destinado a mostrar com clareza como a falta de segurança em uma corporação pode afetar drasticamente no decorrer dos negócios, também será evidenciado as tecnologias à serem aplicadas no decorrer deste trabalho.

2 REVISÃO DE LITERATURA

2.1 Redes de computadores

Uma rede de computadores existirá quando houver certa quantidade de nós interligados entre si e onde seja possível haver troca de informações e/ou compartilhamento de recursos. Um nó é qualquer dispositivo interligado à rede: pode ser um computador, um hub ou switch, um notebook, uma impressora entre outros (CARMONA;HEXSEL, 2007).

As redes de computadores têm crescido explosivamente. Há duas décadas, poucas pessoas tinham acesso a uma rede. Agora, a comunicação via computador transformou-se em uma parte essencial da infraestrutura de todos (COMER, 2007).

2.2 Breve Histórico

As redes de computadores surgiram na necessidade de interligar computadores, e para que houvesse a troca de informações entre os mesmos. A troca de informação é de extrema importância e atualmente faz parte do modelo estrutural de qualquer corporação.

A maior parte das pessoas conhece a internet por meio de suas aplicações: a World Wide Web, e-mail, redes sociais online, streaming de áudio e vídeo, mensagens instantâneas, compartilhamento de arquivos, para citar apenas alguns exemplos. Isso quer dizer que interagimos com a internet como usuários da rede.

Os usuários da internet representam a maior classe de pessoa que interagem com a Internet de alguma maneira, mais existem vários outros grupos importantes. Existe o grupo de pessoas que criam as aplicações, um grupo que se expandiu bastante nos últimos anos, quando plataformas de programação poderosas e novos dispositivos, como smartphones, criaram novas oportunidades para desenvolver aplicações rapidamente e levá-las para um grande mercado. Depois existem aqueles que operam ou administram as redes, em sua maior parte, uma tarefa feita nos bastidores, mais fundamental e normalmente muito complexa.

Com a prevalência das redes domésticas, mais e mais pessoas também estão se tornando mesmo que de uma forma limitada, operadores de rede. Finalmente, existem aqueles que projetam e constroem os dispositivos e protocolos que coletivamente compõem a internet (PETERSON; DAVIE, 2013).

2.3 Usos de redes de computadores

As redes de computadores têm por seu objetivo fazer com que sistemas se comuniquem diretamente através de comunicação de dados e dessa forma interligam as tecnologias de computadores para se comunicarem entre si.

As redes de computadores surgiram da necessidade da troca de informações, onde é possível ter acesso a um dado que está fisicamente localizado distante de você (TORRES, 2001).

A fusão dos computadores e das comunicações e telecomunicações influenciaram diretamente na forma como os computadores são atualmente organizados. O modelo de um único computador realizando todas as tarefas requeridas não existe mais e está sendo substituído pelas redes de computadores, nas quais os trabalhos são realizados por vários computadores separados, interconectados por alguma via de comunicação (AMARAL, 2012).

> "Se você tem dois computadores isolados num mesmo ambiente, estes funcionam, mas não conversam. Não batem papo. Talvez tivessem muitas coisas para combinar, mas como nunca se conheceram, não poderão ser amigos. Não temos uma rede ainda, pois os computadores não trocam informações" (MENDES, 2009, p. 10).

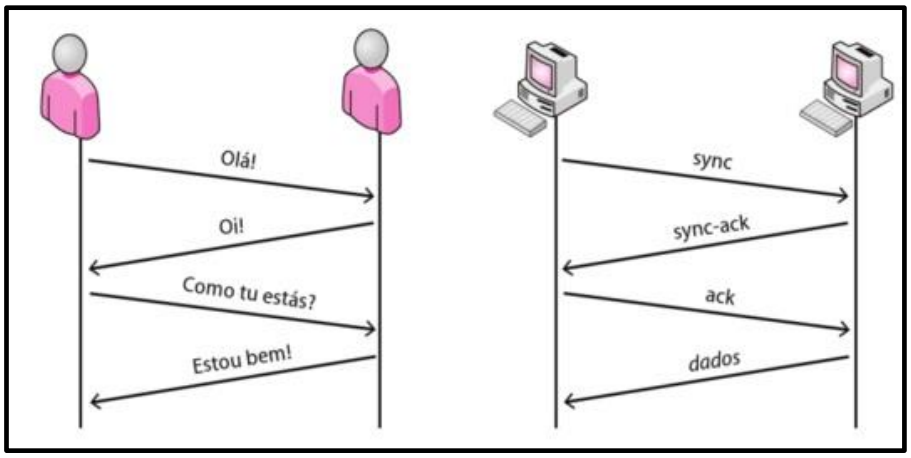
2.3.1 Protocolos de comunicação

Os protocolos de comunicação são responsáveis por transportar um determinado dado de um ponto da rede até outro, feito a solicitação o protocolo responde.

No nosso relacionamento em sociedade, utilizamos diariamente o protocolo de boas maneiras, como responder "tudo bem" "ou mais ou menos" quando alguém nos pergunta "como tu estás? ". Este é o conceito de protocolo, uma predefinição de mensagens e respostas, que podem ser utilizadas tanto por pessoas como por computadores para a realização de uma comunicação, conforme podemos observar na figura 1 (SCHMMITT et al., 2013).

Desta forma a analogia humana se baseia no princípio de pergunta e resposta, é desta maneira que os protocolos de rede se comunicam, a única parte que os diferenciam são que os componentes de software se comunicam com os de hardware.

Figura 1: Protocolos de Comunicação



Fonte: Schmmitt et al., 2013.

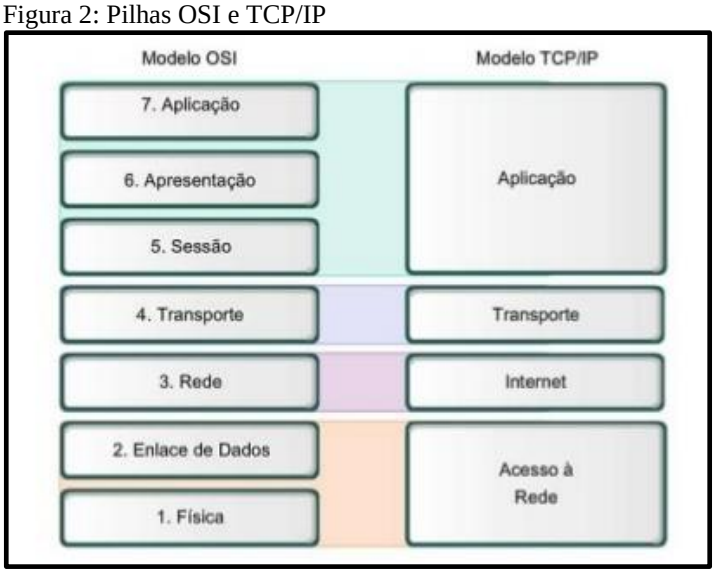
2.3.2 Protocolo TCP/IP

Segundo (ALBUQUERQUE, 2001) **“**O nome TCP/IP se deve a dois dos principais protocolos na família: o Transmission Control Protocol (TCP) e o Internet Protocol (IP). A família de protocolos TCP/IP é organizada em quatro camadas: interface com a rede, internet, transporte e aplicação figura 2”. Os protocolos dessa família encontram-se descritos em documentos chamados Request for Comments (RFC), que podem ser obtidos pela Internet.

O protocolo IP tem, entre as suas responsabilidades, de rotear os dados entre máquina de origem e a máquina de destino e faz parte da camada internet. O TCP, por sua vez, é um dos protocolos na camada de transporte. Os protocolos figura 2 nessa camada são fim a fim, podem ser ou não orientados a conexão, podem garantir ou não a entrega dos dados e possibilitam a troca de informações entre processos nas máquinas envolvidas na comunicação. Os processos enviam e recebem dados através de serviços providos pelos protocolos dessa camada.

Outro importante protocolo da família TCP/IP é o User Datagram Protocol (UDP). O UDP presta serviço não orientado a conexão e não garante a entrega dos dados no destino, diferente do TCP, que presta um serviço orientado a conexão e que tenta garantir a entre dos dados ao destino. O protocolo de transporte a ser utilizado depende das características da aplicação (ALBUQUERQUE, 2001).

Figura 2: Pilhas OSI e TCP/IP



Fonte: Albuquerque, 2001.

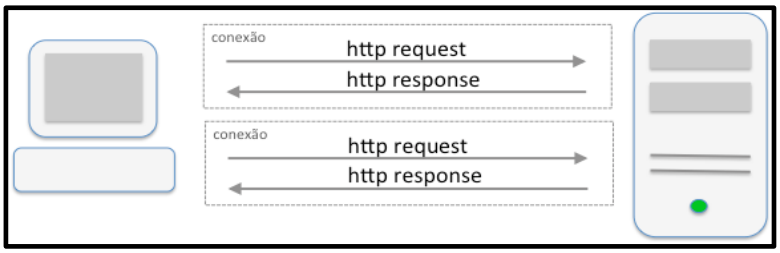
2.3.3 HTTP

O protocolo HTTP é a comunicação entre o cliente e o servidor, é enviado uma requisição do cliente e o servidor enviar uma resposta com a solicitação.

HTTP é implementado em dois programas: um programa cliente e o outro servidor. Os dois programas, executados em sistemas finais diferentes, conversam um com o outro por meio de troca de mensagens HTTP (KUROSE, 2009), analisando esta afirmação o protocolo HTTP basicamente estrutura como deve ser está comunicação e as trocas de mensagens.

De acordo com Kurose (2009) o protocolo HTTP usa o protocolo TCP como seu protocolo de transporte, ou seja, quando um usuário requisita uma página Web, primeiramente o HTTP inicia uma conexão TCP com o servidor, assim estabelecida, o browser envia mensagens de requisições HTTP para os objetos da página. O servidor recebe essas requisições figura 3 e responde essas mensagens de resposta HTTP que contém os objetos.

Figura 3: Requisição HTTP



Fonte: http://www.devmedia.com.br/crie-aplicacoes-com-websocket/29055

2.4 IP (INTERNET PROTOCOL)

O protocolo IP é um dos protocolos mais importantes para a da família TCP/IP, esse protocolo foi introduzido na ARPANET no início dos anos 80, e tem sido utilizado juntamente com o TCP desde então. A principal característica desse protocolo é que a transmissão é efetuada sem a necessidade de uma conexão entre máquina fonte máquina destino, sendo baseada no envio de datagrama que podem passar por muitas redes intermediárias até chegarem ao destino. Um datagrama IP consiste de um cabeçalho e uma área de dados (TANENBAUM, 2003).

2.5 DNS (Domain Name System)

Segundo Alecrim (2005) DNS é a sigla para Domain Name System (Sistema de Nomes de Domínio). Ele é um recurso usado em redes TCP/IP (o protocolo utilizado na internet e na maioria das redes) que permite acessar computadores sem que o usuário ou sem que o próprio computador tenha conhecimento de seu endereço IP.

2.6 DHCP - Dynamic Host Configuration Protocol

O DHCP é a abreviatura de Dynamic Host Configuration Protocol que é um serviço utilizado para automatizar as configurações do protocolo TCP/IP nos dispositivos de rede. O protocolo é responsável por atribuir o endereço IP, máscara de rede, gateway padrão e servidor DNS, servidor WINS para os dispositivos de rede poderem ser utilizado. O DHCP cliente ou servidor é hoje encontrado em uma variada gama de plataformas como o Unix, o Windows, o Linux, entre outros (ALECRIM, 2005).

2.7 Protocolo ARP

De acordo com (FOROUZAN, 2009) o protocolo ARP está ligado à associação do IP ao endereço físico dos computadores em rede, entretanto, quando o endereço de IP é reconhecido por uma de suas interfaces de rede o protocolo ARP atua como identificador de endereços físicos, quando a internet é reconhecida o ARP é responsável para encontrar o endereço físico do nó.

2.6 ICMP – Internet Control Message Protocol

Segundo Tanenbaum (2003) Todos os hosts TCP/IP implementam o ICMP. As mensagens do ICMP são carregadas nos datagramas IP e são usadas para enviar mensagens de erro e de controle.

2.6.1 Características de um sistema inseguro

A segurança de sistemas existe por um conjunto de fatores. Engana-se quem pensa que somente por utilizar uma plataforma Unix ao invés do Windows está seguro. Ou que é só colocar um antivírus e um firewall na sua empresa que está tudo bem (ASSUNÇÃO, 2002). De acordo com o autor a proporção deste problema é muito maior, pois a geralmente temos vários fatores em comum.

Administrador é o ponto chave para qualquer sistema de computador é o administrador. Ele é responsável por fazer como que tudo funcione perfeitamente. Checa os dados, administra usuários, controle servidores, checa logs, tudo todos os dias (ASSUNÇÃO, 2002). A partir desta de afirmação, verificamos que mesmo que se o administrador esteja usando uma plataforma com a segurança pré-definida, para obter o seu perfeito funcionamento é preciso verificar falhas, e assim que possível corrigir de forma continua.

2.6.2 Firewall

Como introdução, podemos afirmar que Firewall é um programa que detém autonomia concedida pelo próprio sistema para pré-determinar e disciplinar todo o tráfego existente entre o mesmo e outros hosts/redes; salvo situações onde o Firewall é um componente de soluções denominado “Firewall-in-a-box”, onde neste caso, trata-se não tão somente de um software de um software e sim de um agrupamento de componentes incluindo software e hardware, ambos projetados sob medida para compor soluções de controle perante o tráfego de um host/rede (URUBATAN NETO, 2004).

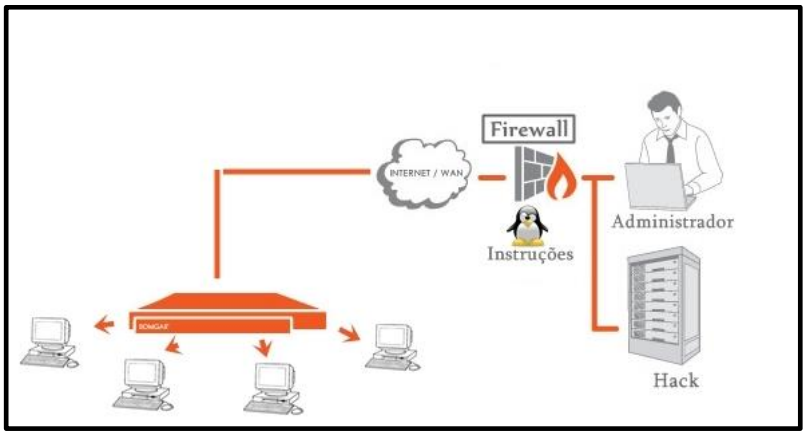
De acordo com Kurose (2006) um firewall é a combinação de software e hardware que isola uma rede local de uma empresa da internet, controlando os pacotes que podem ou não trafegar entre as duas redes. O firewall permite que o administrador da rede controle o acesso entre a rede que administra e o mundo externo.

Analisando está afirmação um firewall deve intermediar uma determinada rede e com isso deverá analisar quais os pacotes e qual o seu destino, a partir dessa ideia, regras serão aplicadas de modo que se determine o destino do trafego.

O objetivo de um firewall é gerenciar as comunicações que ocorrem para dentro e para fora de uma determinada rede (CARISSIMI et al., 2009).

Um firewall segue as regras, diretrizes previamente configuradas pelo administrador de rede que são chamadas de políticas de segurança. Ele normalmente é instalado logo após o *link* da *internet*, a figura 4 ilustra a maneira que o *firewall* pode ser projetado antes da rede interna seja construída.

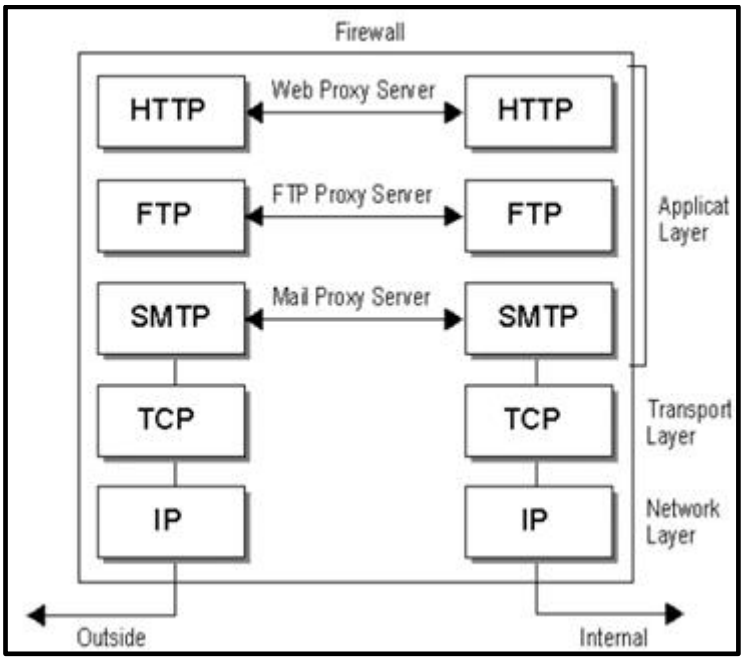
Figura 4: Estrutura Firewall



Fonte: Arquivo do autor

A finalidade do firewall é formar uma linha vedada de defesa, onde sua razão vem para proteger os bens internos de uma empresa, portanto, o firewall pode ser visto como um processo de funcionamento: um existe para bloquear trafego e o outro existe para permiti-lo. Qualquer das duas estratégias é baseada em um plano total se segurança de uma empresa (BAQUI, 2012).

Figura 5: Um firewall Típico



Fonte: SENNA JUNIOR, 2008.

2.7 Tipos de firewalls

Existem basicamente dois tipos de firewalls:

Nível de aplicação: Este tipo de firewall analisam o conteúdo do pacote para tomar suas decisões de filtragem. Firewalls deste tipo são mais intrusivos (pois analisam o conteúdo de tudo que passa por ele) e permitem um controle relacionado com o conteúdo do tráfego. Alguns firewalls em nível de aplicação combinam recursos básicos existentes em firewalls em nível de pacotes combinando as funcionalidades de controle de tráfego/controle de acesso em uma só ferramenta. Servidores proxy, como o Squid, são um exemplo deste tipo de firewall.

Nível de pacotes: Este tipo de firewall toma as decisões baseadas nos parâmetros do pacote, como porta/endereço de origem/destino, estado da conexão, e outros parâmetros do pacote. O firewall então pode negar o pacote (DROP) ou deixar o pacote passar (ACCEPT). O iptables é um excelente firewall que se encaixa nesta categoria.

Os dois tipos de firewalls podem ser usados em conjunto para fornecer uma camada dupla de segurança no acesso as suas máquinas/máquinas clientes.

2.7.1 IPTABLES

O código iptables é um firewall a nível de pacotes e funciona baseado no endereço/ porta de origem/destino do pacote, prioridade, etc. Ele funciona através da comparação de regras para saber se um pacote tem ou não permissão para passar. Em firewalls mais restritivos, o pacote é bloqueado e registrado para que o administrador do sistema tenha conhecimento sobre o que está acontecendo em seu sistema.

O iptables é um código de firewall das versões 2.4 do kernel, que substituiu o ipchains (presente nas séries 2.2 do kernel). Ele foi incluído no kernel da série 2.4 em meados de Junho/Julho de 1999. A história do desenvolvimento (desde o porte do ipfw do BSD para o Linux até o iptables (que é a quarta geração de firewalls do kernel) está disponível no documento, Netfilter-howto (SILVA, 2010).

Ainda de acordo com (SILVA, 2010) abaixo segue as características do código iptables na implementação.

- Especificação de portas/endereço de origem/destino
- Suporte a protocolos TCP/UDP/ICMP (incluindo tipos de mensagens icmp)
- Suporte a interfaces de origem/destino de pacotes
- Manipula serviços de proxy na rede
- Tratamento de tráfego dividido em chains (para melhor controle do tráfego que entra/sai da máquina e tráfego redirecionado.
- Permite um número ilimitado de regras por chain
- Muito rápido, estável e seguro
- Possui mecanismos internos para rejeitar automaticamente pacotes duvidosos ou malformados.
- Suporte a módulos externos para expansão das funcionalidades padrões oferecidos pelo código de firewall
- Suporte completo a roteamento de pacotes, tratadas em uma área diferente de tráfegos padrões.
- Suporte à especificação de tipo de serviço para priorizar o tráfego de determinados tipos de pacotes.
- Permite especificar exceções para as regras ou parte das regras
- Suporte a detecção de fragmentos
- Permite enviar alertas personalizados ao syslog sobre o tráfego aceito/bloqueado.
- Redirecionamento de portas Masquerading

- Suporte a SNAT (modificação do endereço de origem das máquinas para um único IP ou faixa de IP's).
- Suporte a DNAT (modificação do endereço de destino das máquinas para um único IP ou fixa de IP's)
- Contagem de pacotes que atravessaram uma interface/regra
- Limitação de passagem de pacotes/conferência de regra (muito útil para criar proteções contra, syn flood, ping flood, DoS etc).

2.7.2 Filtragem de pacotes

Kurose (2006) define que as regras de filtragens especificas pelo administrador são aplicadas, após a análise, dos cabeçalhos de datagramas determinando se o datagrama será descartado ou não.

Ainda segundo KUROSE (2006) as decisões de filtragem são baseadas em:

- Endereço IP
- Porta TCP e UDP de origem e de destino
- Tipo de mensagem ICMP
- Datagramas de inicialização de conexão usando bits TCP SYN ou ACK.

O Firewall quando filtro de pacotes, possui a capacidade de analisar cabeçalhos (Headers) enquanto os mesmos trafegam. Mediante esta análise, que é fruto de uma extensa separação de regras (URUBATAN NETO, 2004).

2.7.3 Objetos da Linguagem e definição o código

Todavia, será a plataforma Linux. Sendo assim, será adotado o filtro de pacotes iptables, aplicativo padrão em distribuições Linux para a construção de *Firewalls*. O iptables é disponibilizado sob Licença Pública Geral e foi desenvolvido em sua maior parte pela equipe do núcleo Netfilter, mas recebeu contribuições de outros desenvolvedores (NETFILTER, 2010). Os objetos básicos e os tipos de dados da linguagem foram baseados na forma de definição de regras do Iptables.

O nome Iptables vem de três tabelas padrão que estão listadas na figura 6 Cada interface no seu sistema pode ter seus pacotes administrados e modificados pelas correntes contidas em cada uma dessas tabelas (STANGER; LANE, 2002).

Figura 6: Tabela Filter

Nome da tabela	Corrente padrão	Permite a filtragem dos pacotes
Filter	INPUT	Permite a filtragem dos pacotes
	FORWARD	
	OUTPUT	
Nome da tabela	Corrente padrão	Descrição
Nat	PREROUTING	Permite o mascaramento
	OUTPUT	
Mangle	POSTROUTING	
	OUTPUT	

Fonte: Stanger;Lane, 2002.

2.8 Tabela Filter

Tabela filter que é o conjunto de regras com finalidades gerais, como bloquear, negar, realizar logs. As regras existentes nesta tabela não têm poder de alterar as configurações dos pacotes. Basicamente todas as regras de filtragem estão nesta tabela, pois ela é de uso geral.

As 3 (três) possíveis chains da tabela filter são:

• INPUT: Pacotes cujo destino final é a própria máquina firewall.

• OUTPUT: Pacotes que saem da máquina firewall.

• FORWARD: Pacote que atravessa a máquina firewall, cujo destino é uma outra máquina. Este pacote não sai da máquina firewall e sim de outra máquina da rede ou fonte. Neste caso a máquina firewall está repassando o pacote (REHEM, 2010).

2.9 Tabela Nat

As regras da tabela nat tem o poder de alterar características de origem ou de destino de um pacote. Como característica de origem entende-se IP de origem ou porta de origem e como características de destino tem-se o IP destino e porta destino. A tabela nat possui (BAQUI, 2012).

Ainda (BAQUI, 2012) cita 3 (três) conjuntos de regras:

• PREROUTING: Tratamento do pacote antes de ele ser roteado.
• POSTROUTING: Tratamento dado ao pacote após ele ser roteado.
• OUTPUT: Pacotes que saem do roteador.

Note que nas regras acima, temos mais duas ações a SNAT e DNAT:

• SNAT: É utilizada quando queremos alterar o endereço de origem do pacote.

Somente a chain POSTROUTING pode ser usada na ação SNAT.

• DNAT: É utilizada quando desejamos alterar o endereço de destino do pacote.

Está ação é aplicada para fazer redirecionamento de portas, redirecionamento de servidor, load balance e proxy transparente. As chains que podem ser utilizadas para esta ação são PREROUTING e OUTPUT.

• REDIRECT: Pode ser utilizada para fazer redirecionamento de portas. Quando fazemos um redirecionamento de portas usamos o dado --to-port após a ação REDIRECT.

Analisando as características as chains são de modo locais onde as regras definidas pelos usuários são armazenadas para a operação do firewall. Desta maneira, as chains estão relacionadas às tabelas que serão usadas.

3 REGRAS DE FIREWALL

De acordo com URUBATAN NETO (2004) as principais políticas de ações são:

• ACCEPT: Onde todo e qualquer acesso é liberado.

• REJECT: Onde todo e qualquer acesso é bloqueado, gerando uma mensagem de retorno.

• DROP: Onde todo e qualquer acesso é bloqueado, porém ele não gera nenhuma resposta (exceto para localhost).

• LOG: Cria um log referente à regra em /var/log/messages.

As regras de firewall geralmente são compostas de uma Tabela, Opção, Chain, Dados e Ação. Através destes elementos podemos especificar o que fazer com os pacotes.

iptables [- t tabela] [opção] [chain] [dados] -j [ação]

Exemplo:

iptables –A FORWARD –d 192.168.1.1 –j DROP

Tabela: Filter (é a default)

Opção: -A

Chain: FORWARD

Dados: -d 192.168.1.1

Ação: DROP

3.1 Demonstrando IPTABLES/NETFILTER

A partir de regras de firewall podemos definir regras de aceitação e negação, vejamos abaixo um exemplo de aceitação de acesso total com a regra ACCEPT supondo que se esteja utilizando o IP 192.168.100.0/24 e local host 127.0.0.1.

Figura 7: Exemplo de aceitação Iptables

```
#Permite pingar da rede interna para qualquer um dos servidores externos.
iptables-A OUTPUT-p icmp - icmp-type echo-request-j ACCEPT
iptables-A INPUT-p icmp - icmp-type echo-reply-j ACCEPT

#Permite conexão SSH criptografada somente na rede interna com IP 192.168.100.0/24
iptables-A OUTPUT-o eth0-p tcp-d 192.168.100.0/24 - dport 22-m
state - Estado Novo, ESTABLISHED-j ACCEPT iptables -A INPUT -i
tcp eth0-p - sport 22 -m state - state ESTABLISHED
-j ACCEPT

#Acesso total para localhost 127.0.0.1
iptables -A INPUT -s 127.0.0.1 -j ACCEPT
iptables -A INPUT -s 127.0.0.1 -j ACCEPT
```

Fonte: Arquivo do Autor.

3.2 NMAP – Network Mapper

É uma ferramenta muito poderosa, criada por Gordon Fyodor Lyon em 1997, é amplamente utilizada por profissionais de TI para realizar varredura de porta, descoberta de serviço e detecção de versões. A ferramenta possui uma série de funções específicas para a realização de diferentes varreduras de rede (GIAVAROTO; SANTOS, 2012).

Abaixo podemos demonstrar como utilizar a ferramenta Nmap para descobrir vulnerabilidades no servidor figura 8 e como corrigir tais falhas com as regras de scripts iptables.

Podemos dar exemplos da análise da rede em primeiro momento para verificar as possíveis vulnerabilidades presentes no ambiente. Feito uma pesquisa das portas abertas e dos resultados apresentados pelo Nmap, utilizando principalmente uma melhor familiarização com a ferramenta.

Figura 8: Comando Nmap Análise de Vulnerabilidade

Fonte: Arquivo do Autor

Na demonstração acima o exame analisou 1000 portas do computador e foi verificado que vinte delas estão respondendo as requisições. Dessas 14 estão abertas, e podem ser possíveis formas de acesso indevido ao sistema.

3.2.1 Importância na utilização de Firewalls

A capacidade de conectar qualquer computador em qualquer lugar a qualquer outro computador em qualquer lugar é uma faca de dois gumes. É muito divertido para as pessoas navegarem pela Internet quando estão em casa. Para os gerentes de segurança

das empresas, trata-se de um pesadelo. Muitas empresas têm grandes quantidades de informações confidenciais on-line, segredos comerciais, planos de desenvolvimento de produtos, estratégias de marketing, análises financeiras etc.

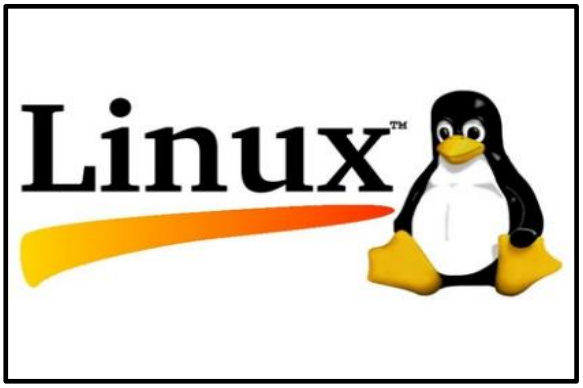
A revelação dessas informações para um concorrente poderia ter terríveis consequências. Além do perigo das informações virem a público, também há o perigo do vazamento dessas informações dentro da empresa. Em particular, vírus, vermes e outras pestes digitais podem burlar a segurança, destruir dados valiosos e consumir muito tempo dos administradores, que tentam eliminar a confusão causada por eles. Com frequência, eles são trazidos por funcionários descuidados que querem brincar com algum jogo novo muito divertido.

Em consequência disso, são necessários mecanismos para manter os "bons" bits e descartar os "maus" bits. Um dos métodos é usar o IPsec, que protege os dados em trânsito entre sites seguros. No entanto, o IPsec não faz nada para impedir as pestes digitais e os intrusos de invadirem a LAN da empresa. Para ver como alcançar esse objetivo, precisamos examinar os firewalls (TANENBAUM, 2003).

3.2.2 Unix e GNU/Linux

O Linux é um sistema operacional de código fonte aberto, derivado do Unix e poderoso o suficiente para ser adotado em servidores do mundo todo para as mais diversas tarefas (GOMES JUNIOR, 2007).

Figura 9: Logo Linux



Fonte: http://cdn-3.famouslogos.us/images/linux-logo.jpg

Basicamente, uma Distribuição Linux (ou simplesmente distro) é composta do kernel Linux, ferramentas GNU e um conjunto variável de software dependendo de seus propósitos.

Carlos Marimoto, explica que empresas de grande porte como IBM e Sun ao adotarem ao sistema operacional em seus produtos não são somente pelo argumento de redução de custos, mais sim por trazer tranquilidade de utilização, portanto a questão está sendo levado a sério em diversos âmbitos.

> "No geral o sistema se tornou bastante profissional, maduro o suficiente para tornar-se uma opção viável ao Windows para empresas e usuários domésticos, não apenas no velho argumento do custo, mas por realmente ter qualidade. É interessante perceber que além de empresas como a IBM e Sun, que estão adotando o Linux em grande escala em seus produtos, tivemos a participação até mesmo da Microsoft na Linux World de 2002, mostrando que até mesmo eles estão levando o Linux a sério. Como dizia Mahatma Ghandi: "primeiro eles te ignoram, depois riem de você, então finalmente resolvem te enfrentar e aí você vence. " (MARIMOTO, 2004).

3.2.3 Vantagens de utilizar Linux como ferramenta de segurança

Para determinamos qual ferramenta de segurança é de maior eficiência na área segurança computacional, antes tudo precisamos determinar qual a melhor plataforma para implantação de segurança.

A grande demanda de servidores com a plataforma Linux e a alta no crescimento da utilização ao redor do mundo é resultado de inúmeras vantagens que o sistema operacional tem sobre os outros sistemas (ASSUNÇÃO, 2002) cita algumas dessas vantagens, como a de distribuição gratuita, criptografia de senhas inquebráveis (somente descobertas através de várias tentativas de erros) possui as melhores ferramentas de rede e faz melhor gerenciamento de permissões (GOMES JUNIOR, 2007) também cita algumas das inúmeras vantagens na migração e utilização da plataforma Linux, diversos pontos devem ser analisados, como também o custo, compatibilidade de software, melhor aproveitamento do hardware e suporte.

As duas ferramentas serão criadas e testadas no S.O Ubuntu Linux, podendo ser desenvolvidas em outras distribuições, baseado no mesmo âmbito como o Debian, Red Hat, Fedora, CentOS.

Figura 10: Logo Ubuntu

Fonte: http://design.ubuntu.com/wp-content/uploads/ubuntu-logo14.png

3.4 Shell Script

O Shell Script é um poderoso interpretador de comandos do Linux, o conhecimento de Shell Script é essencial para quem deseja torna-se um razoável administrador, mesmo que não pense em desenvolver scripts. Como em tempo de boot uma máquina Unix/Linux executa os diversos scripts contidos em etc/rc.d para configurar o sistema e inicializar os serviços, um entendimento detalhado destes é importante para a análise do comportamento do sistema, e sua possível modificação (NEVES, 2010).

> "Programar em shell geralmente envolve a manipulação de texto e o gerenciamento de processos e de arquivos. As tarefas complexas ficam com as ferramentas do sistema como grep, sed, dd e find que se encarregam dos bits e bytes e possuem interface amigável via opções de linha de comando. Além de possuir as funcionalidades básicas de uma linguagem estruturada normal e a integração natural com o sistema operacional e suas ferramentas, há as facilidades de redirecionamento, em que é possível combinar vários programas entre si, multiplicando seus poderes e evitando reescrita de código" (JARGAS, 2008, p. 26)

Podemos dizer que o shell script poderoso interpretador via linha de comandos trabalhando diretamente com o sistema operacional que quando digitado traduz para que o kernel do Linux compreenda.

3.5 Benefícios de implementação de um servidor Proxy

Um servidor proxy é um estágio intermediário entre os servidores de redes diferentes ou separadas, como uma rede local (LAN) e a Internet. É usado para fazer o armazenamento de dados para certos serviços e para controle de segurança e administrativo (STANGER; LANE, 2002).

3.6 SQUID

Squid é um proxy-cache de alta performance para clientes web, suportando FTP, gopher e HTTP. O Squid mantém meta de dados e especialmente objetos armazenados na RAM, cacheia buscas DNS e implementa cache negativo de requests falhos. Ele suporta SSL, lista de acesso complexas e logging completo. Por utilizar o Internet Cache Protocol, o Squid consiste em um programa principal, squid, um sistema de busca e resolução de nomes dnsserver e alguns programas adicionais para reescrever requests, fazer autenticação e gerenciar ferramentas de clientes (BASTOS, 2008).

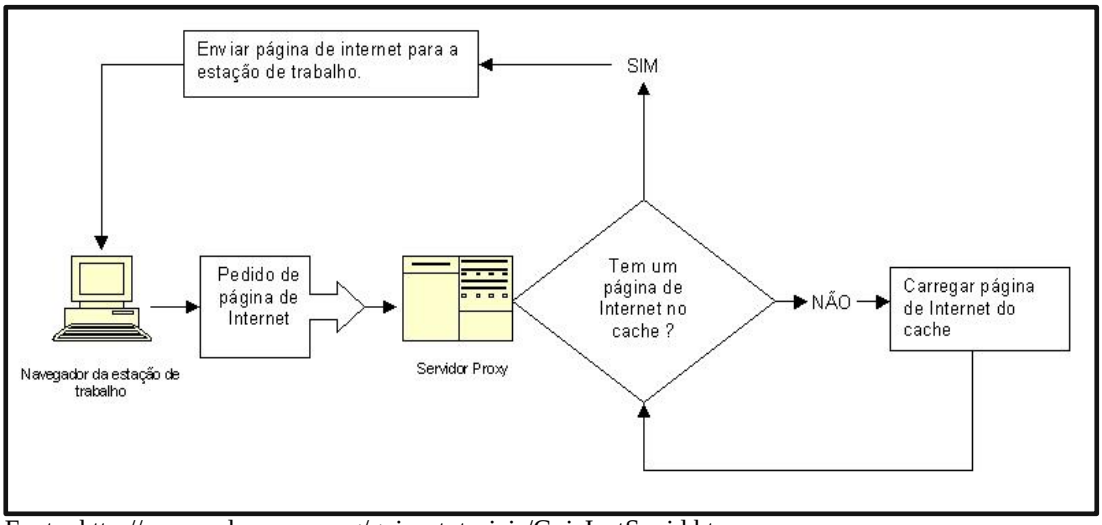
Segundo BASTOS (2008), os servidores Proxy possuem as seguintes características:

a) **Velocidade de acesso:** Ganho de velocidade com armazenamento local, evitando novo tráfego externo;
b) **Disponibilidade:** Deve estar a maior parte do tempo disponível. Em grandes organizações, deve-se extinguir falhas, e manter sempre em funcionamento, utilizando-se de redundância de servidores;
c) **Transparência e Ostensividade:** Utiliza políticas de utilização, controle de acesso e recolhe informações sobre o tráfego de dados;
d) **Simplicidade:** Servidor configurado de forma a facilitar sua administração, assim não dependendo de uma única pessoa para lidar com os problemas;
e) **Capacidade de trabalhar com redes heterogêneas:** Devem funcionar em multiplataformas e atender bem os usuários de ambas

O Squid é constantemente aperfeiçoado, melhorando seu desempenho e adicionando novas funcionalidades. Ele se demonstra extremamente estável quando levado ao limite (REHEM, 2010).

O Proxy aguarda uma requisição interna (Firewall, Rede interna), verifica se ele tem armazenado esta solicitação em cache, se está, está disponível, responde para o cliente a resposta, se não, passa está para o servidor remoto (Rede Externa), recebe a resposta armazena as informações em cache e envia para estação cliente que solicitou.

Figura 11: Fluxo das solicitações dos servidores proxy



Fonte: http://www.mlaureano.org/guias_tutoriais/GuiaInstSquid.htm

As principais vantagens de um Proxy Squid são o armazenamento de conteúdo em cache, otimizando muito utilização do link de internet, definir regras para acesso a URLs, controlo de acesso IP.

3.7 ACL

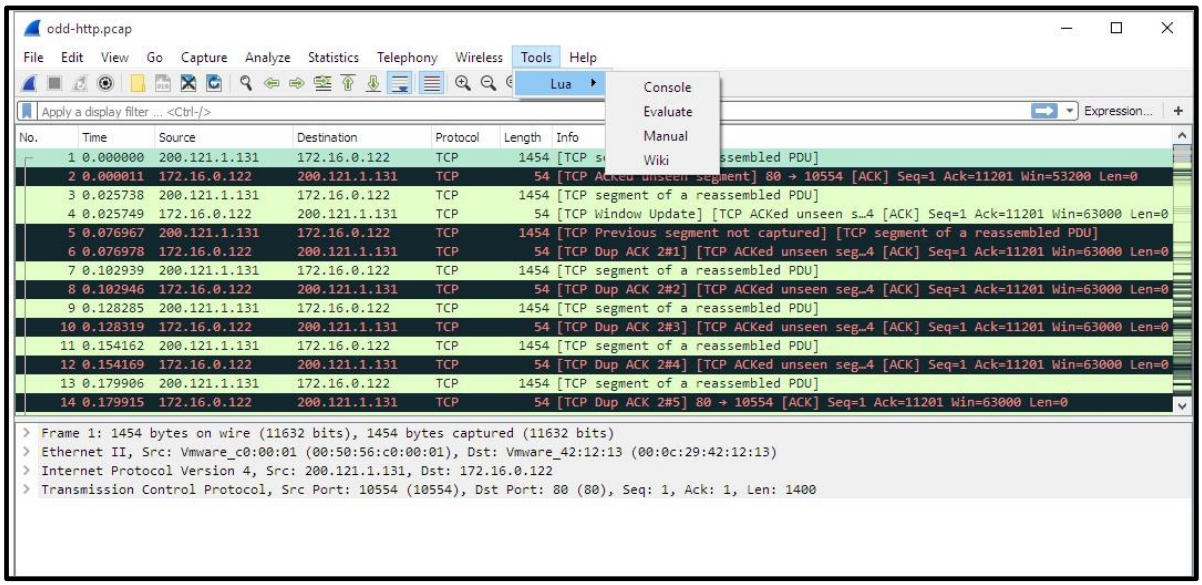
O Squid possui um eficiente mecanismo de controle dos acessos de internet numa rede. Através deste mecanismo, torna-se possível a criação de listas altamente customizadas, capazes de filtrar desde simples domínios até conteúdos mais específicos. (MORIMOTO, 2004) as ACL's - (Access Control Lists) ou listas de controle de acesso, constituem a grande flexibilidade e eficiência do Squid. Através delas podem-se criar regras para controlar o acesso à Internet das mais diferentes formas. Praticamente todo o processo de controle do Squid é feito com o seu uso. O uso das listas de controle de acesso é a parte mais importante da configuração de um servidor proxy Squid, pois se bem

configuradas podem trazer um nível de segurança muito bom para a rede. Entretanto se mal configuradas podem ter o resultado oposto, já que além da falsa sensação de segurança não será aproveitada a principal funcionalidade do Squid.

3.8 Wireshark

O wireshark é uma ferramenta de análise de pacotes de rede, ele é capaz de capturar pacotes em uma rede e exibi-los de forma detalhada através de vários filtros já disponíveis na ferramenta e usando filtros criados pelo usuário, conforme figura 12, o wireshark é disponibilizado na licença GNU, ou seja, é um software open-source, disponível em várias plataformas (Windows, Mac, Linux, BSD).

Figura 12: Wireshark

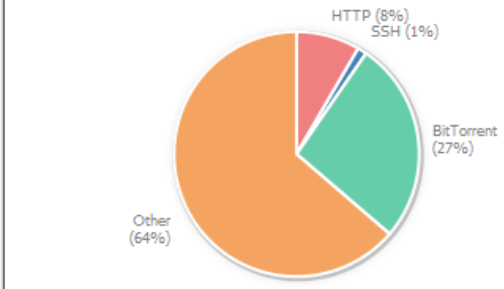


Fonte: http://www.wireshark.org

3.8.1 Ntop

O Ntop é um software que possui métodos capazes de detectar pacotes que estão sendo transmitidos na rede e segmenta-los de acordo com características que possam ser analisadas, atualmente possuindo opções de medida de trafego, monitoramento de trafego, planejamento e otimização da rede, etc. Essas características fazem do Ntop uma ferramenta amplamente capaz de auxiliar no desenvolvimento e manutenção das mais diversas redes de computadores.

Figura 13: Comunicação entre protocolos

TCP/UDP Protocol	Data	Flows	Accumulated Percentage / Historical Protocol View
FTP	5.9 KBytes	31	0%
HTTP	30.0 MBytes	13,841	8.4%
DNS	2.7 MBytes	27,694	0%
NBios/CIFS	19.8 KBytes	188	0%
Mail	0.1 KBytes	2	0%
NFS/AFS	17.3 KBytes	30	0%
VoIP	0.2 KBytes	2	0%
X11	5.0 KBytes	45	0%
SSH	4.3 MBytes	33	1.2%
eDonkey	497.8 KBytes	54	0%
BitTorrent	95.3 MBytes	349	26.8%
Messenger	3.4 KBytes	21	0%
Other TCP/UDP-based Protocols	223.0 MBytes	514,370	62.7%
Accumulated View			

Fonte: Arquivo do Autor

3.8.2 SSH

O Secure Shell (SSH) é um protocolo para conectar-se remotamente a computadores pela porta 22 de uma maneira segura. Para conectar a partir de um cliente no servidor, o serviço de ssh permite fazer o acesso remoto ao console de sua máquina, em outras palavras, você poderá acessar sua máquina como se estivesse conectado localmente ao seu console (substituindo o rlogin e rsh). A principal diferença com relação ao serviço telnet padrão, rlogin e rsh é que toda a comunicação entre cliente/servidor é feita de forma encriptada usando chaves públicas/privadas RSA para criptografia garantindo uma transferência segura de dados (MILLER, 2002).

Podemos afirmar que em conexões sem criptografia (rsh, rlogin) os dados trafegam de forma desprotegida e caso exista algum sniffer instalado em sua rota com a máquina destino, tudo o que fizer poderá ser capturado (incluindo senhas).

3.8.3 PHP e Bootstrap

A linguagem PHP foi concebida para ser utilizada exclusivamente na Web e implementa todas as estruturas de programação, manipulação de dados, suporte para a implementação das estruturas de dados (SICA, 2011).

Figura 14: PHP

Fonte:http://www.weblizards.com.br/wp-content/uploads/2014/08/php.png

A linguagem PHP foi escolhida para criar interface web do software (Squid Web Proxy Shell) com o intuito de fazer com que o haja interação e flexibilidade de uso por parte do usuário.

Utilizado a tecnologia responsiva do framework de CSS Twitter Bootstrap, além de outros recursos presentes em sua folha de estilos. É importante ressaltar, que assim como Linux e o Shell Script, o Bootstrap e PHP também são free, portanto ferramentas não pagas que permitem livre alteração de código.

Figura 15: Bootstrap

Fonte: http://m5designstudio.com/wp-content/uploads/2013/04/bootstrap_responsive_layout.png

3.8.4 MYSQL

O MySQL é um SGBD extremamente versátil, usado para os mais diversos fins. Permite acesso a seu banco de dados a partir de scripts em PHP, através de um aplicativo desenvolvido em C ou C++, ou praticamente qualquer outra linguagem (MORIMOTO, 2009).

Por conter uma fácil integração com GNU/Linux, Apache e PHP, formadores do conjunto LAMP, MySQL foi o SGBD escolhido para armazenamento de dados da aplicação Squid Web Proxy Shell.

3.9 Apache

O Apache é um servidor web open source, sendo a principal tecnologia da Apache Software Foundation. Ele encontra-se disponível em diversas plataformas de sistemas operacionais, como Windows, Novell Netware, OS/2 e diversos outros do padrão POSIX (Unix, Linux, FreeBSD, etc.).

Segundo MORIMOTO (2009) também destaca sobre a qualidade dos servidores apache:

> A principal característica do Apache é a modularidade. Ao invés de ser um aplicativo grande e complexo, que tenta desempenhar sozinho todas as funções, o Apache se limita a executar uma única tarefa: entregar páginas html e outros tipos de arquivos aos clientes. Qualquer outra coisa é invariavelmente feita por um módulo externo (MORIMOTO, 2009, p.351).

Desta forma o autor destaca a maneira como o apache trabalha a sua modularidade e divisão de tarefas se tornando assim um servidor web robusto e de qualidade.

4 SEGURANÇA COMPUTACIONAL

Durante as primeiras décadas de sua existência, as redes de computadores foram usadas principalmente por pesquisadores universitários, com a finalidade de enviar mensagens de correio eletrônico, e também por funcionários de empresas, para compartilhar impressoras. Sob essas condições, a segurança nunca precisou de maiores cuidados. Porém, como milhões de cidadãos comuns atualmente estão usando as redes para executar operações bancárias, fazer compras e arquivar sua devolução de impostos, a segurança das redes está despontando no horizonte como um problema potencial (TANENBAUM, 2003).

4.1 A arquitetura de segurança OSI

Para avaliar efetivamente as necessidades de segurança de uma organização e avaliar e escolher diversos produtos e políticas de segurança, o gerente responsável precisa de algum meio sistemático de definir os requisitos de segurança e caracterizar as técnicas para satisfazer esses requisitos

A recomendação X.800 da ITU-T² Security architecture for OSI, define tal técnica sistemática. A arquitetura de segurança OSI é útil para os gerentes como um meio de organizar a tarefa de prover segurança. Além disso, como essa arquitetura foi desenvolvida como padrão internacional, fornecedores de computador e comunicação desenvolveram recursos de segurança para os seus produtos e serviços, que se relacionam com essa definição estruturada de serviços e mecanismos.

Para os nossos propósitos, arquitetura de segurança OSI oferece uma visão geral útil, abstrata, de muitos dos conceitos dos quais este livro trata. A arquitetura de segurança OSI enfoca ataques, mecanismos e serviços, que se relacionam com essa definição estruturada de serviços e mecanismos

Para os nossos propósitos, a arquitetura OSI oferece uma visão geral útil, abstrata, de muitos dos conceitos dos quais este livro trata. A arquitetura de segurança OSI enfoca ataques, mecanismos e serviços de segurança. Este podem ser definidos resumidamente da seguinte forma (STALLINGS, 2008).

a) **Ataque à segurança:** Qualquer ação que comprometa a segurança da informação pertencente a uma organização.

b) **Mecanismo de segurança:** Um processo (ou um dispositivo incorporado tal processo) que é projetado para detectar, impedir ou permitir a recuperação de um ataque à segurança.

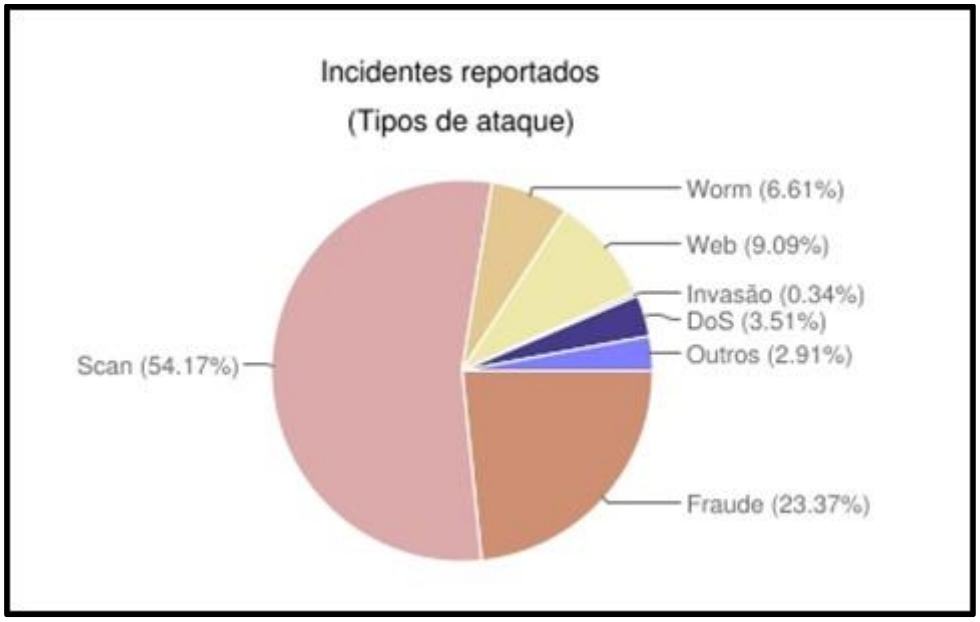
c) **Serviços de segurança:** Um serviço de processamento ou comunicação que aumenta a segurança dos sistemas de processamento de dados e as transferências de informação de uma organização. Os serviços servem para frustar ataques à segurança e utilizam um ou mais mecanismos de segurança para prover o serviço (STALLINGS, 2008).

Analisando este assunto podemos ver que se definirmos os mecanismos de defesa seguindo um padrão ou arquitetura podemos organizar as tarefas como queremos atingir um alto nível em relação segurança computacional.

4.2 Ataques à segurança

Segundo web site (CERT.BR, 2015) (*Centro de Estudos, Resposta a Tratamento de Incidentes de Segurança no Brasil*), os incidentes reportados no ano de 2015 de janeiro a dezembro tem aumentado de forma surpreendente, vejamos a seguir o gráfico aonde mostra de forma equalizada os dados reportados pela pesquisa da CERT.

Figura 15: Incidentes Reportados ao CERT.br – janeiro a dezembro de 2015.



Fonte: CERT.br, 2015.

Figura 16: Invasões 2015.

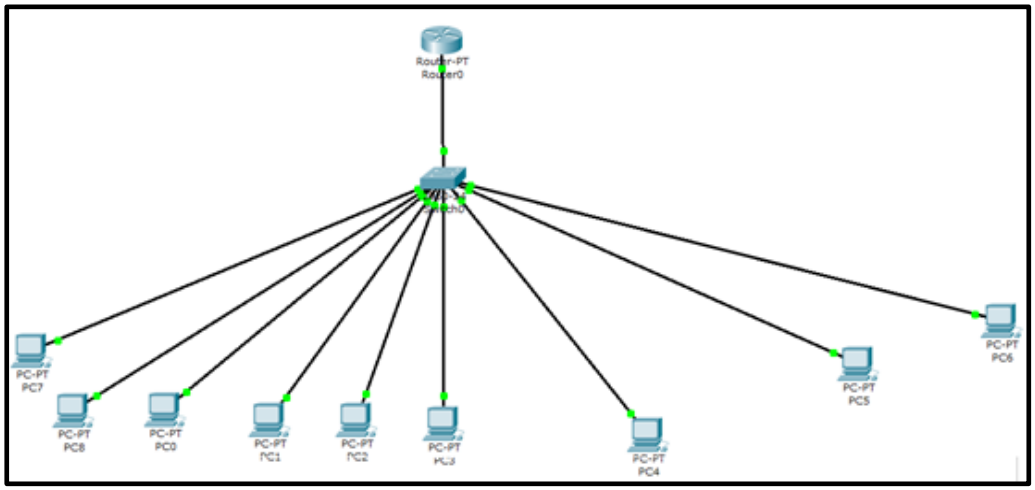
a) Worm: notificações de atividades maliciosas relacionadas com o processo automatizado de propagação de códigos maliciosos na rede.
b) dos (DoS - *Denial of Service*): notificações de ataques de negação de serviço, onde o atacante utiliza um computador ou um conjunto de computadores para tirar de operação um serviço, computador ou rede.
c) Invasão: um ataque bem-sucedido que resulte no acesso não autorizado a um computador ou rede.
d) Web: um caso particular de ataque visando especificamente o comprometimento de servidores Web ou desfigurações de páginas na Internet.
e) Scan: notificações de varreduras em redes de computadores, com o intuito de identificar quais computadores estão ativos e quais serviços estão sendo disponibilizados por eles. É amplamente utilizado por atacantes para identificar potenciais alvos, pois permite associar possíveis vulnerabilidades aos serviços habilitados em um computador.
f) Fraude: segundo Houaiss, é "qualquer ato ardiloso, enganoso, de má-fé, com intuito de lesar ou ludibriar outrem, ou de não cumprir determinado dever; logro". Esta categoria engloba as notificações de tentativas de fraudes, ou seja, de incidentes em que ocorre uma tentativa de obter vantagem.
Outros: notificações de incidentes que não se enquadram nas categorias anteriores.

Fonte: CERT.BR (2015)

4.2.1 Antigo ambiente Colégio Maria Imaculada

A figura 17 apresenta a estrutura física atual da rede do Colégio Maria Imaculada mostrando um breve mapeamento da estrutura usada para comunicação entre os setores e de comunicação está sendo utilizada. Como podemos ver na figura 17, os pontos que interligam e forma a comunicação de rede do colégio é feita somente por equipamento distribuição no caso switch, no entanto, essa estrutura não dispõe de equipamento para formar um ponto do meio da rede.

Figura 17: Ambiente CMI

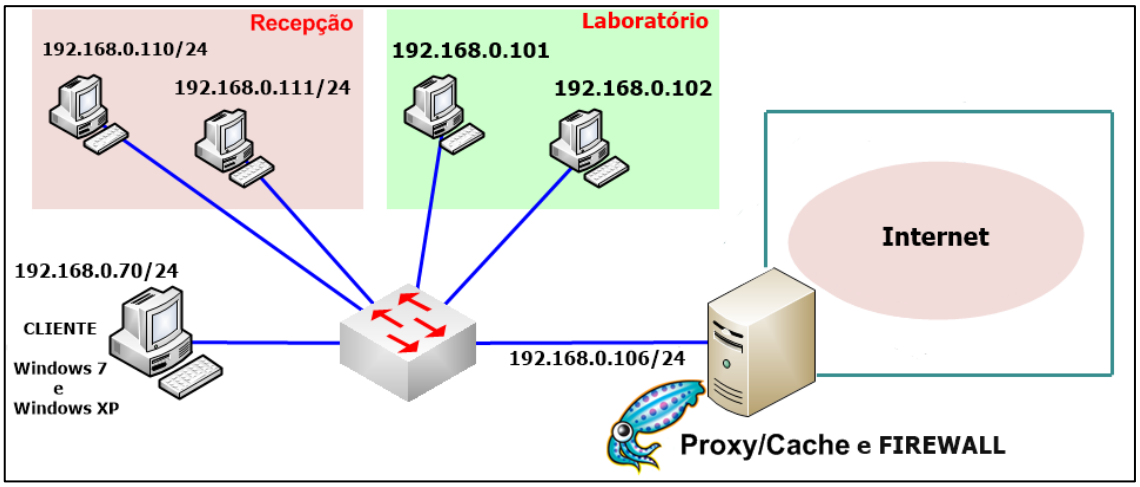


Fonte: Arquivo do Autor

4.2.2 Correções realizadas

Os grandes problemas de segurança nas empresas são de aspecto técnico onde as tecnologias estão ligadas diretamente na internet, desta forma com o Colégio Maria Imaculada não é diferente um ataque a um de suas estações de trabalho pode trazer significativos prejuízos financeiros e estruturais, de acordo com o próprio autor durante entrevistas com a superintendente do colégio em 2016 foram evidenciando várias falhas de segurança, pois a mesma não tem equipamentos suficiente para deter quaisquer outros tipos de ataques e invasões de nível mais sofisticado, outro problema estava relacionado ao laboratório de alunos aonde não se tinha controle de acesso dos alunos a URLs de web sites, na figura 18, um breve diagrama de como a estrutura ficou após a implantação do servidor Linux.

Figura 18: Nova Estrtutura CMI



Fonte: Arquivo do Autor

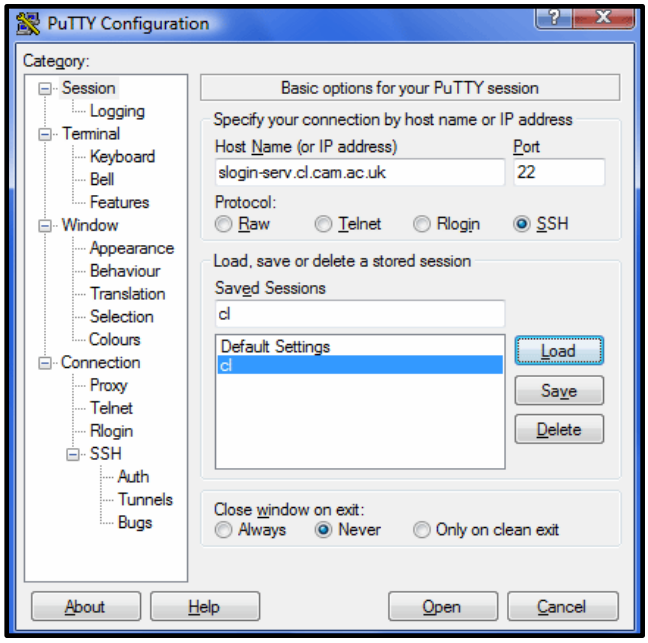
5 FERRAMENTAS DO PROJETO

Este tópico mostra a descrição das ferramentas que serão utilizadas para auxiliar no desenvolvimento deste projeto.

5.1 Putty

Está sendo utilizado o software cliente Putty para fazer conexão remota segura SSH pela desenvolvido em ambiente Windows, ele possibilita comunicação criptografada entre cliente e servidor através da porta 22.

Figura 19: Putty



Fonte: Arquivo do Autor

5.2 Análise de tráfego com Nmap

O Nmap é uma ferramenta extremamente útil para procurar hosts, portas abertas, versões de software, sistemas operacionais, hardware e fragilidades, geralmente mapeando a superfície de ataque da rede. Ele é útil em cada etapa dos testes de penetração, identificando os componentes conectados ao entrar em um novo segmento de rede.

A ferramenta Nmap é tão poderosa que é utilizada tanto por administradores de rede, quanto por criminosos. Ambos trabalhando para analisar e descobrir vulnerabilidades nos sistemas pesquisados (BEZERRA, 2012).

6 METODOLOGIA

6.1 Documentação

Em pesquisa bibliográfica este trabalho abrange toda bibliografia já tornada pública em relação ao tema estudado, desde publicações avulsas, boletins, jornais, revistas, livros, pesquisas, monografias, teses, materiais cartográficos, etc. E sua finalidade é colocar o pesquisador em contato direto com tudo o que foi escrito, dito ou filmado sobre determinado assunto (LAKATOS; MARCONI, 2001).

6.2 Natureza da Pesquisa

Podemos definir as informações que foram apresentadas neste trabalho como uma pesquisa de caráter exploratório e teórico. O objetivo de uma pesquisa exploratória é familiarizar-se com um assunto ainda pouco conhecido, pouco explorado.

Desta forma estará apto a construir hipóteses. Como qualquer exploração, a pesquisa exploratória depende da intuição do explorador (neste caso, da intuição do pesquisador). Por ser um tipo de pesquisa muito específica, quase sempre ela assume a forma de um estudo de caso (GIL, 2008).

6.3 Tipo da Pesquisa

Esta pesquisa é de caráter bibliográfica e de campo com a finalidade acerca de um problema específico para o qual se procura uma solução ou resposta.

Pesquisa bibliográfica é a obtenção de estudos e dados, tudo que esteja relacionado com tema aqui proposto. É coletado e apanhado trabalhos que já foram realizados para objeto de estudos como publicações, livros, monografias, pesquisas, teses, etc.

Pesquisa de campo evidencia fatos que a partir de um determinado problema consegue se informações e coleta de dados referente ao tema e assunto abordado (MARCONI; LAKATOS, 2003).

6.4 Técnicas da Pesquisa

Podemos definir esta pesquisa como bibliográfica e de campo, um caso particular da pesquisa qualitativa.

A pesquisa bibliográfica servirá, como primeiro passo, para se saber em que estado se encontra atua1mente o problema, que trabalhos já foram realizados a respeito e quais são as opiniões reinantes sobre o assunto. Como segundo passo, permitirá que se estabeleça um modelo teórico inicial de referência, da mesma forma que auxiliará na determinação das variáveis e elaboração do plano geral da pesquisa (MARCONI; LAKATOS, 2003)

6.5 Coleta de dados

Para a coleta de dados foram utilizadas pesquisas bibliográficas com o intuito de reunir o máximo de informações sobre o assunto adotado nessa modalidade de pesquisa.

De forma geral, qualquer informação publicada (impressa ou eletrônica) é passível de se tornar uma fonte de consulta. Os livros constituem-se nas principais fontes de referências bibliográficas em guias publicados em livros adotados como referências em sistemas formais de ensino constituem-se em um conhecimento pronto para a consulta (MARCONI; LAKATOS, 2003).

7 PROJETO

7.1 Hardware

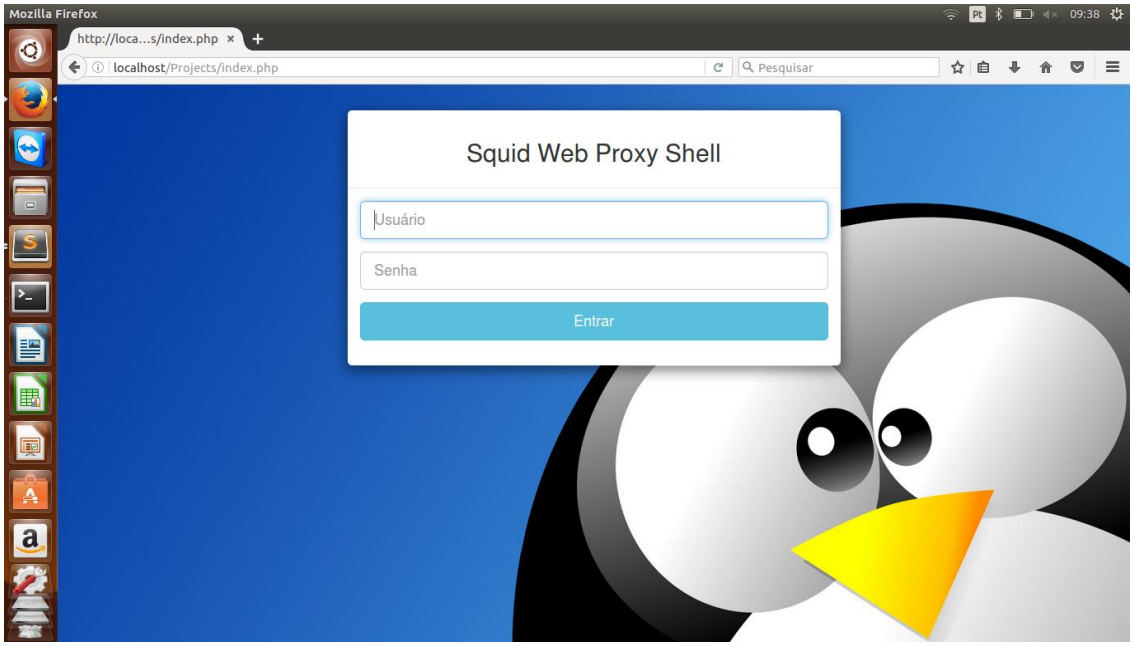
Visto que este projeto foi implantado de forma definitiva, foi realizado em duas etapas a primeira foi a elaboração do projeto, segundo a implantação física do servidor no local e desenvolvimento do software seguido dos testes de usabilidade por parte do gerente do laboratório e os demais usuários.

Na primeira etapa o desenvolvimento do projeto será hospedado em um servidor com sistema operacional *Linux Debian versão 7*, *processador core I5*, 4 GB de memória RAM e capacidade de armazenamento de 1 TB, será colocado um computador com configuração robusta de forma que fique disposto a novos upgrades de software ou periféricos atualizados.

7.2 Interfaces Squid Web Proxy Shell

A intenção do Squid Web Proxy Shell é que o usuário possa utilizar o software de forma que não precise necessariamente o administrador de rede para manter as regras do sistema de autenticação de usuários e sim qualquer pessoa com conhecimento básico possa bloquear sites por palavras de restrição, por endereço de IP destino ou por grupo de usuários liberados e grupos de usuários bloqueados. Abaixo a tela inicial do Squid Web Proxy Shell.

Figura 20: Tela Inicial

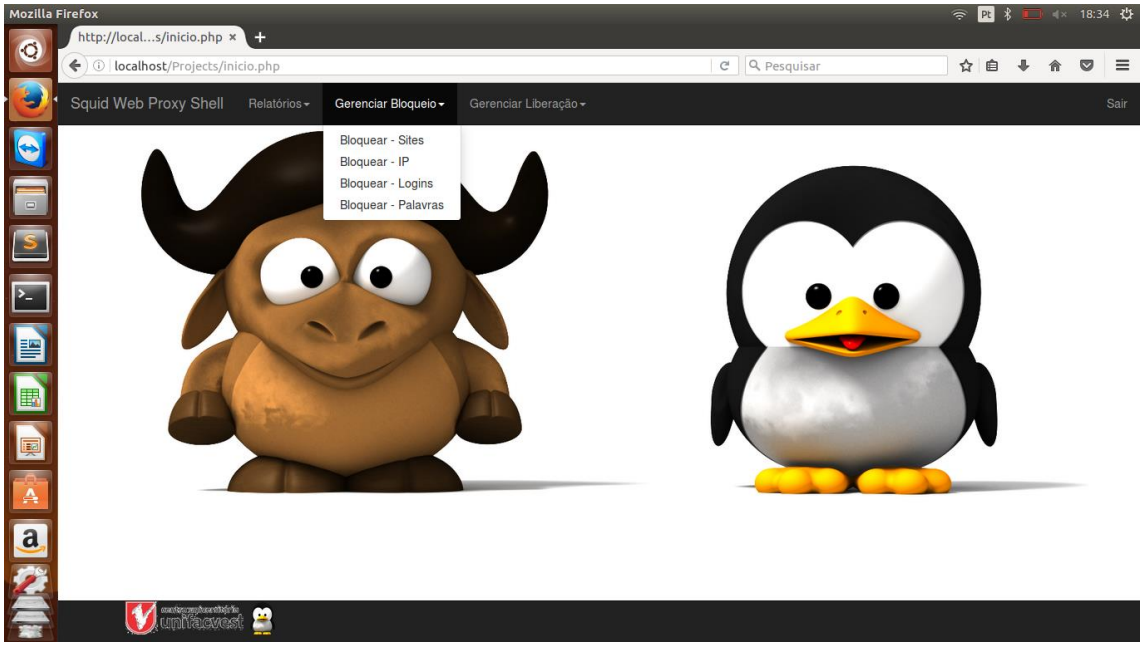


Fonte: Arquivo do Autor

7.2.1 Projeto

A intenção do Squid Web Proxy Shell é que o usuário possa utilizar o software de forma que não precise necessariamente o administrador de rede para manter as regras do sistema de autenticação de usuários e sim qualquer pessoa com conhecimento básico possa bloquear sites por palavras de restrição, por endereço de IP destino ou por grupo de usuários liberados e grupos de usuários bloqueados. Abaixo as telas do Squid Web Proxy Shell.

Figura 21: Exemplo de Tela



Terminal Squid Web Proxy Shell

Fonte: Arquivo do Autor

7.2.2 Mensagens ao usuário

Mensagem exibida ao usuário ao passar pela regra do *Squid Web Proxy Shell* após dada a ação pelo usuário, em seguida é exibida a mensagem abaixo no browser, indicando que o bloqueio foi efetuado pelo servidor.

Figura 22: Exemplo de mensagens ao usuário

Fonte: Arquivo do Autor

Fonte: Arquivo do Autor

7.2.3 Menu Ação de Bloqueio

Figura 23: Menu Ação

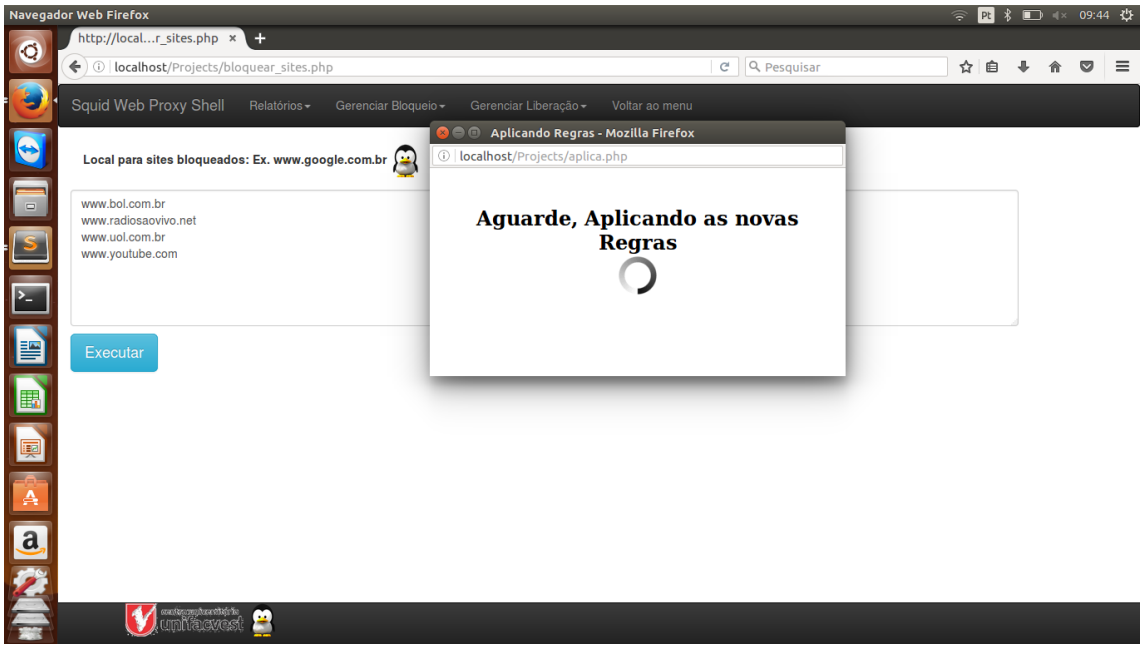
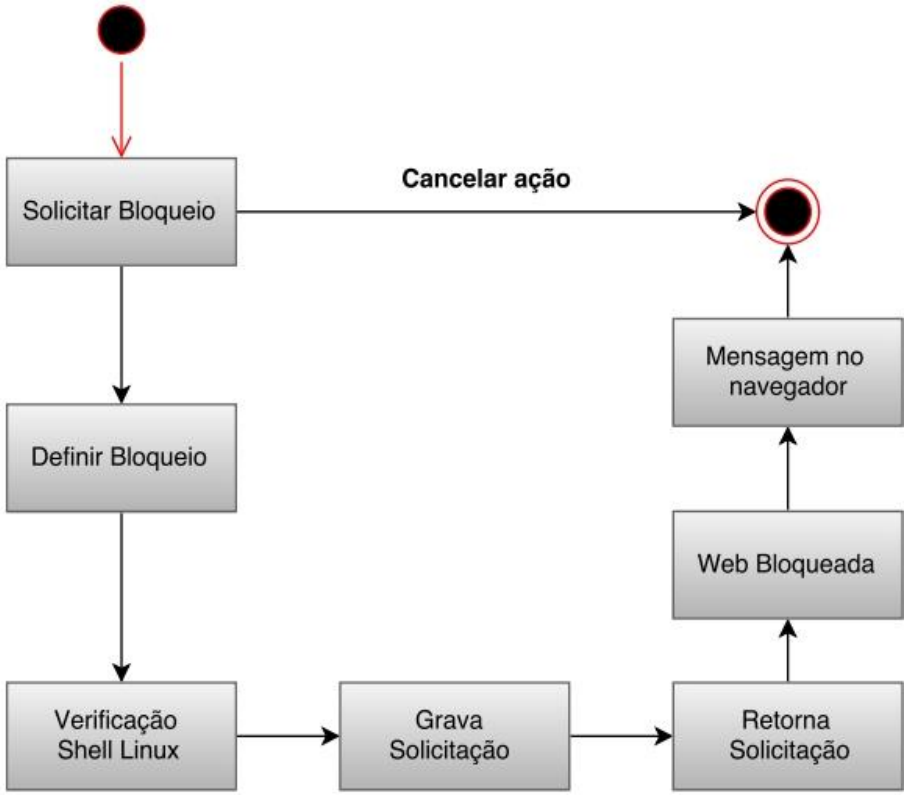


Figura 15: Arquivo do autor

7.2.4 Diagrama de Atividade

Diagrama de atividade demonstram o fluxo de controle de uma determinada tarefa que o sistema irá executar, e de que forma atingirá o objetivo final na execução, abaixo representando pelo diagrama o processo de bloqueio (SILVA;VIDEIRA, 2001).

Figura 24: Diagrama de atividade solicitar bloqueio

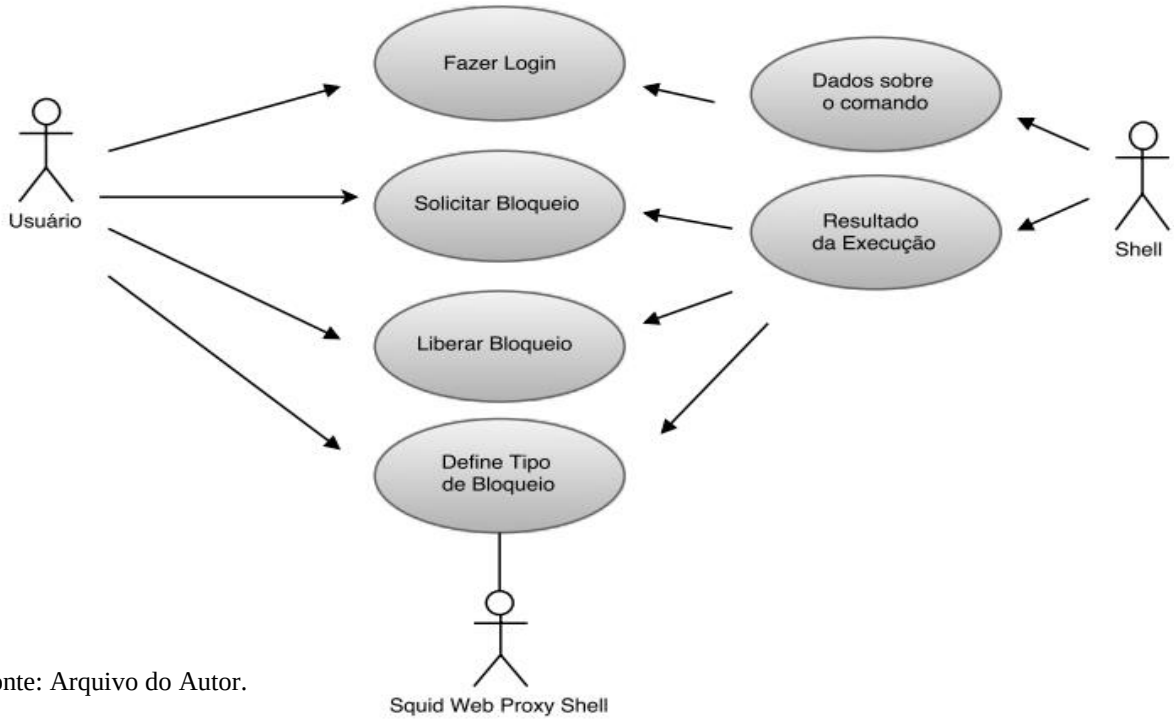


Fonte: Arquivo do Autor

7.2.5 Caso de uso

O diagrama de caso de uso descreve uma sequência de ações que representam um cenário principal e cenários alternativos, com o objetivo de demonstrar o comportamento de um sistema (ou parte dele), através de iterações com atores (MELO, 2010), figura 25 mostra o caso de uso do Squid Web Proxy Shell.

Figura 25: Diagrama de Caso de uso Squid Web Proxy Shell



Fonte: Arquivo do Autor.

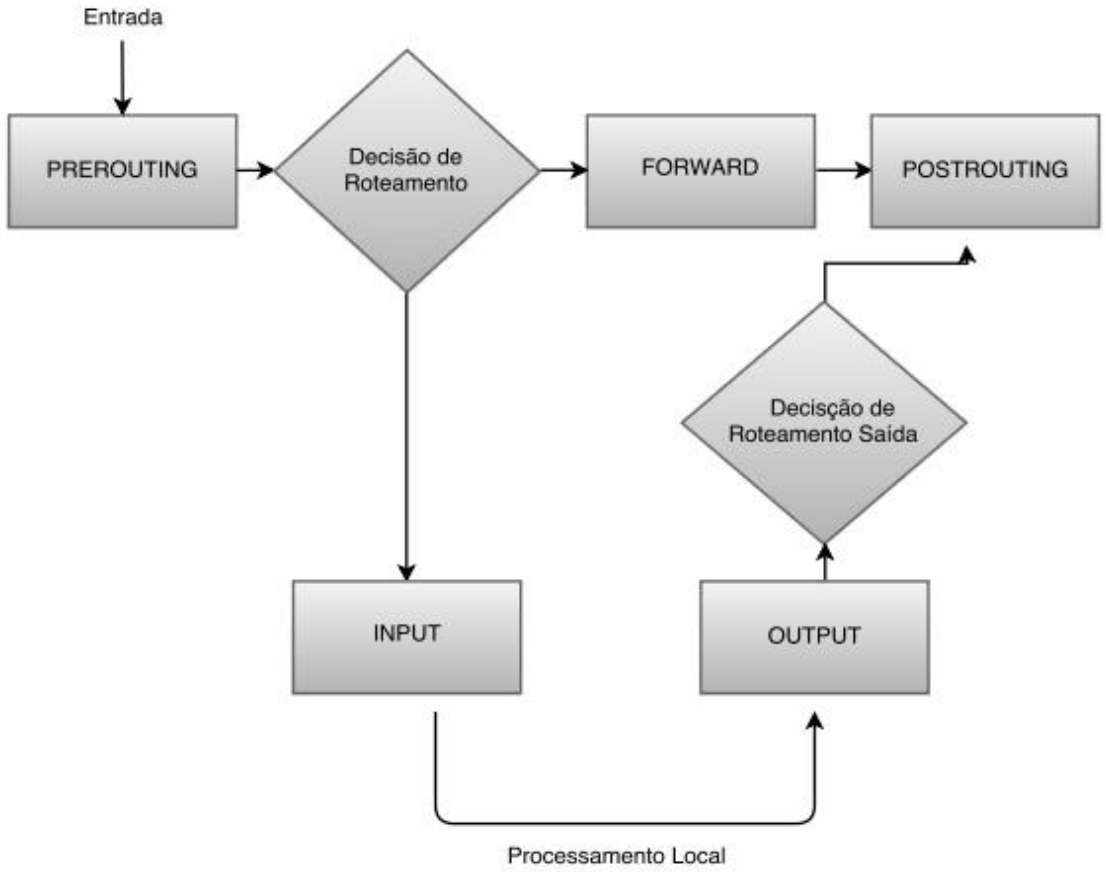
7.2.6 Diagrama de Fluxo Firewall Iptables

Para definir as regras de filtragem é apresentando aqui o digrama de fluxo de dados dos modelos de filtros tais como NAT, masquerading, dentre outras, é necessário saber como funciona o fluxo de pacotes no Iptables. O Iptables lida com o conceito de firewall chains, ou apenas chains, que são as listas de regras nas tabelas de filtragens, NAT e mangle.

O kernel inicia com três chains na tabela “filter”, são elas: INPUT, OUTPUT, FORWARD. Existe também a tabela “NAT”, que lida com as chains PREROUTING, POSTROUTING e OUTPUT, e a tabela “mangle” que lida com as chains PREROUTING e OUTPUT (URUBATAN NETO, 2004).

O diagrama de fluxo descreve o fluxo de informação e as transformações que são aplicadas à medida que os dados se movimentam da entrada para a saída.

Figura 26: Diagrama de Fluxo Firewall Iptables



Fonte: Arquivo do Autor

7.7 CONFIGURAÇÕES DO FIREWALL

Realizadas configurações das regras iptables no linux, ativando os módulos responsáveis pelo seu funcionamento de acordo com nossa necessidade apresentada. Alguns deles vêm ativados por default em várias distribuições GNU/Linux..

O script foi criado para ser executado automaticamente na inicialização do servidor firewall. Salvo com os comandos # iptables-save > /root/regras.fw que encontra-se no APÊNDICE E.

Para uma melhor visualização e entendimento do que foi proposto pelo trabalho, foi criado o arquivo “firewall” na pasta /root, no qual foi dado permissão de execução:

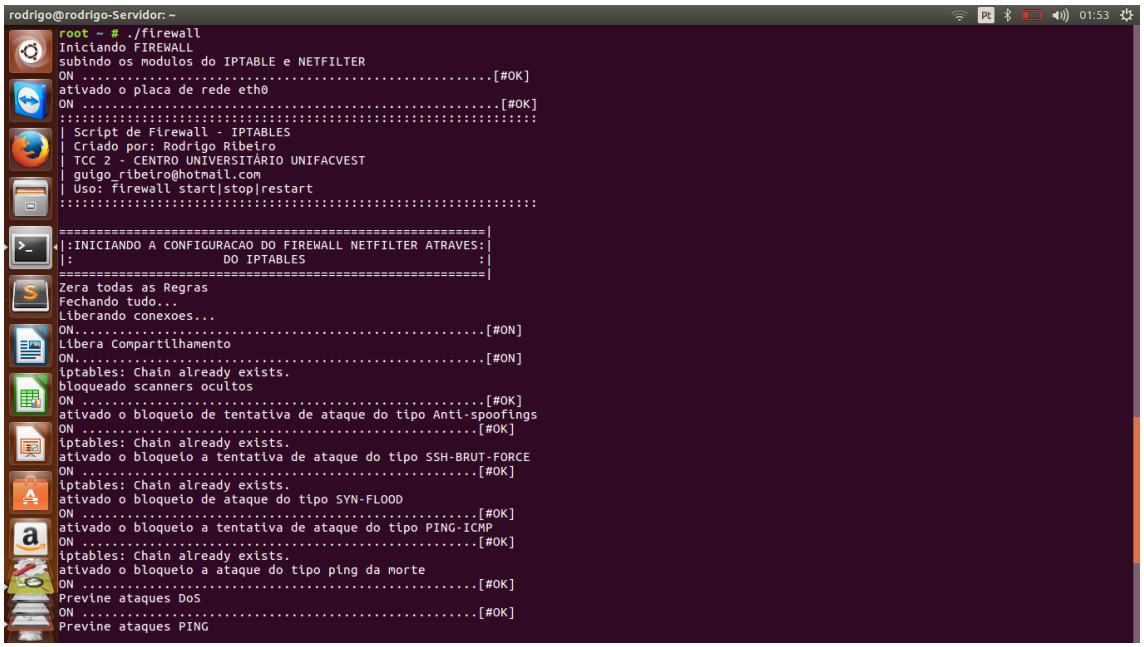
chmod +x firewall

Para fazer funcionar esse script manualmente, executado o seguinte comando

abaixo:

./firewall

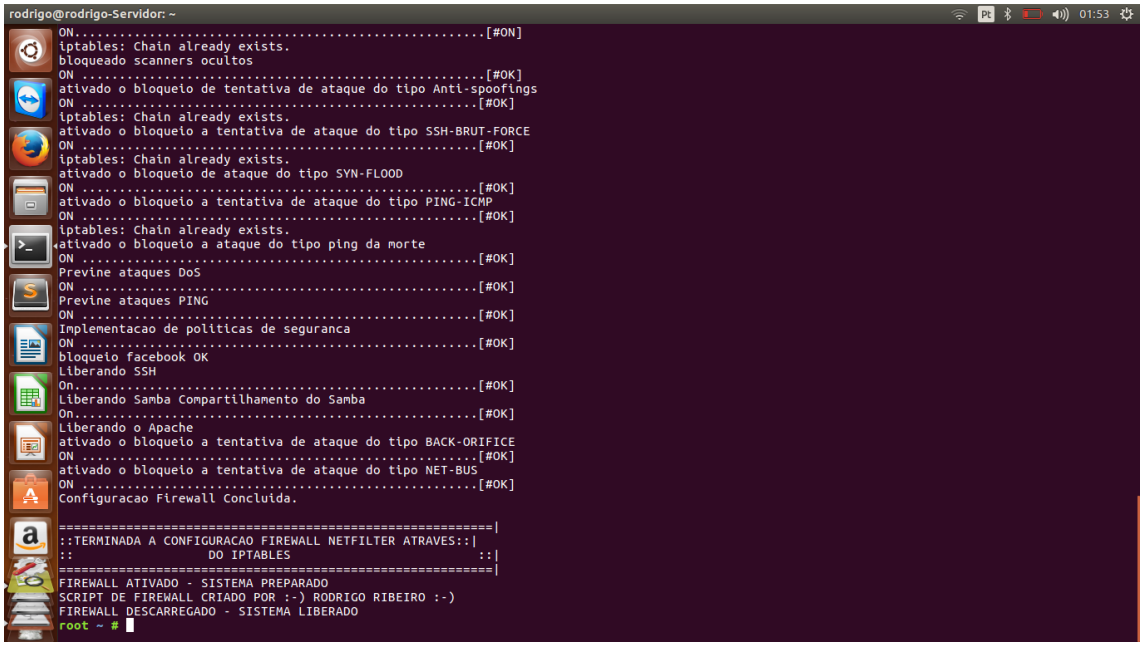
Figura 27: Comando ativar regras ./firewall



Fonte: Arquivo do autor

Com o levantamento de requisitos que foi levantado no exemplo da rede mostrada, as configurações necessárias para se garantir o funcionamento de forma segura, abaixo mensagem de que o firewall foi ativado e está pronto para conexão.

Figura 28: Regras ativadas ./firewall



Fonte: Arquivo do autor.

7.8 Cronograma

O seguinte cronograma foi utilizado para o desenvolvimento deste trabalho.

Figura 29: Cronograma

Atividades	2016										
	Fev.	Mar.	Abr.	Mai.	Jun.	Jul.	Ago.	Set.	Out.	Nov.	Dez.
Determinação do Tema											
Coleta de Dados											
Revisão de Literatura											
Desenvolvimento do Projeto											
Revisão do TCC 1											
Entrega do Trabalho Final											
Definição das funcionalidades do sistema											
Execução do Projeto Software											
Testes											
Implantação											
Entrega do TCC II											
Apresentação a Banca Avaliadora											

Fonte: Arquivo do Autor

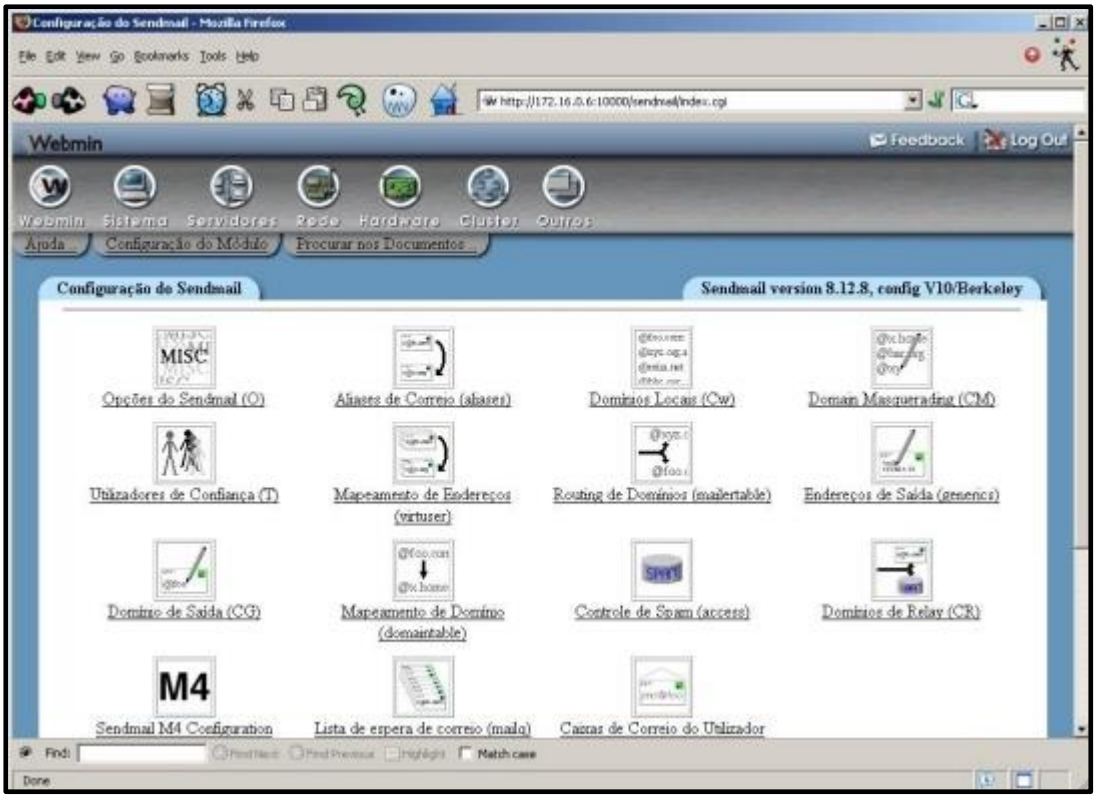
7.9 Trabalhos Correlatos

7.9.1 Webmin

É uma ferramenta de administração gráfica que utiliza linguagem Perl, o webmin funciona como um centralizador de configurações do sistema, monitoração dos serviços e de servidores figura 28, porém tem a sua funcionalidade muito complexa do ponto de vista do usuário, surgindo a necessidade de algo mais amigável e de fácil uso.

O webmin, entretanto, trabalha com uma interface web, ou seja, a possibilidade de se configurar uma máquina através de uma rede, pois basta ter acesso a um navegador. Com isto, é possível configurar uma máquina através de plataforma de hardware e software (CAMERON, 2005).

Figura 30: Interface Webmin



Fonte: https://www.unimep.br/phpg/bibdig/pdfs/2006/LJOVWDFSTNPR.pdf

7.9.2 Limitações Do Projeto

Deve-se destacar que existe uma limitação na abordagem do projeto, levando em conta que será específico para um ambiente, mais que para um futuro projeto deverá atender a demanda de qualquer empresa, podendo ter suas características remodelada conforme a necessidade.

RESULTADOS

Concluiu-se por meio deste estudo e implantação ao Colégio Maria Imaculada, como algo que propiciou a reflexão de uma gama de fatores que devem ser considerados na questão da segurança de redes. Os testes realizados comprovaram a eficiência do aplicativo e iptables que é a interface do usuário com o framework firewall nativo do kernel linux (netfilter) e squid como proxy, dentro dos parâmetros e do nível de complexidade dos testes descritos e tomados como exemplo. E ambas as ferramentas utilizadas proporcionaram uma estratégia de defesa para uma organização que visa evitar as intrusões, bloquear acessos indesejados e controlar a banda de internet.

Todavia, pode-se dizer que a estrutura de um firewall e um servidor proxy com interface se tornou descomplicado e ao mesmo tempo robusto, funcionou muito bem como firewall de rede, suportou mais máquinas e em contrapartida possui facilidade mais elevada nas configurações, obteve melhora no desempenho da internet e diminuição de falhas constantes. Vale ressaltar que o mesmo é o sistema de código aberto mais usado em todo o mundo para defesa de rede.

Por outro lado, Firewall apresenta-se útil para redes menores, de pequenas empresas, suporta muito bem proxy-web, e para surpresa, possui o layout muito intuitivo que torna a configuração mais simplificada.

REFERÊNCIAS

AMARAL, Allan Francisco Forzza. **Redes de computadores** - Colatina: Instituto Federal do Espírito Santo, 2012. p. 17.

ASSUNÇÃO, Marcos Flávio Araújo. **Guia do Hacker Brasileiro**. 1. ed. Florianópolis: Visual Books, 2002.

ALBUQUERQUE, Fernando, **TCP/IP INTERNET: Protocolos & Tecnologias** 3ª Edição – Rio de Janeiro – RJ – Axcel Books 2001. p.19.

ALECRIM, E. **Básico sobre DNS** *(Domain Name System), 2005.* Disponível em http://www.infowester.com/dns.php. Acesso em 05 Maio. de 2016.

BAQUI, Ruben Bambi Tsimba. **Segurança Em Redes Linux Com Firewall, 2012** Disponível em: http://repositorio.roca.utfpr.edu.br:8080/jspui/bitstream/1/1832/1/CT_GESER_II_2012_10.pdf. Acesso em: 21 de maio. 2016.

BASTOS, Eri Ramos. **Manual de Configuração do Squid. (2008)** <http://www.oocities.org/br/dionata.nunes/Documentos/Apostilas/squid.pdf> Acesso em: 21/05/2016.

BEZERRA, Adonel (2012) **"Evitando Hackers"** Ed Ciência Moderna. 1 ed. Rio de Janeiro.

COMER, Douglas E. **Redes de Computadores e Internet:** Abrange Transmissão de dados ligações inter-rede, web e aplicações. Porto Alegre: Bookman. 2007. p. 11.

CAMERON, Jamie – **Webmin Open Country Inc**. - obtido através da Internet. Disponível em: <http://www.webmin.com> Acesso em: junho 2016

CARISSIMI, Alexandre. S.; Rochol, Juergen; Granville, Lisandro. Z. (2009) **Redes de computadores.** Porto Alegre: Bookman. cap. 8, pg. 146-226.

CARMONA, Tadeu. **Universidade Redes.** Digerati Books, 2007, pg. 69.

CGI.BR. **CERT.br registra aumento de ataques de negação de serviço em 2014.** Disponível em: < http://www.cert.br/stats/incidentes/2015-jan-dec/tipos-ataque.html>. Acessado em 06 de maio. 2016.

DANTAS, Marcus Leal. **Segurança da Informação:** Uma abordagem focada em Gestão de Riscos. Olinda - PE: Livro Rápido – Elógica. 2011. p. 11.

DAMMIEN MILLER, **AUUG Winter 2002, SSH tips, tricks & protocol tutorial**, 2002.

FOROUZAN;FEGAN, Behrouz A. com Shopia Chung Fegan; **Protocolo TCP/IP** 3ª ed. - Porto Alegre: AMGH, 2010.

GIL, Antônio Carlos. **Como elaborar projetos de pesquisa.** 5. ed. São Paulo: Atlas, 2008.

GIAVAROTO, S. C. R. **Criando wordlists com o Crunch.** 2012. Disponível em: <http://www.backtrackbrasil.com.br/site/2012/08/criando-wordlists-com-o-crunch/>. Acesso em Maio, 2016.

JARGAS, Aurélio Marinho. **Shell script profissional.** São Paulo: Novatec, 2008.

KUROSE, James F.; **Redes de computadores:** uma abordagem top-down, 3ª ed. São Paulo: Pearson Addsion Wesley. 2006;

MENDES, Gustavo Lopes de Oliveira Santos: **Introdução às Redes de Computadores de Hoje** - Versão BETA 2 - Julho de 2009, p. 19.

MORIMOTO, Carlos, **Entendendo e Dominando o Linux** – Digerati Books, 2004 pg. 44.

MOTA FILHO, João Eriberto. **Descobrindo o Linux: entenda o sistema operacional GNU/Linux**. 3. ed. rev. e ampl. - São Paulo: Novatec Editora, 2012.

MARCONI, Marina de Andrade; LAKATOS, Eva Maria. **Fundamentos de metodologia científica**. 5. ed. São Paulo: Atlas, 2003.

MELO, Ana Cristina. **Desenvolvendo aplicações com UML** 2.2: do conceitual a implantação / Ana Cristina Melo - Rio de Janeiro: Brasport, 2010.

MOTA FILHO, João Eriberto, **Análise de tráfego em redes TCP/IP: utilize tcpdump na análise de tráfegos em qualquer sistema operacional.** São Paulo: Novatec Editora, 2013.

NETO**,** Urubatan. **Dominando Linux Firewall Iptables** – Rio de Janeiro: Ciência Moderna, 2004.

NEVES, Júlio Cezar, **Programação LINUX** / 8ª edições. – Rio de Janeiro: Brasport, 2010.

PETERSON; DAVIE, Larry e Bruce S**. Redes de computadores**: uma abordagem de sistemas – Rio de Janeiro: Elsevier, 2013. Cap. 1, pg. 2.

SCHMITT, Marcelo Augusto Rauh et al. **Redes de computadores: Nível de aplicação e instalação de serviços.** – Dados eletrônicos. – Porto Alegre: Bookman, 2013.

STALLINGS, William. **Criptografia e segurança de redes** – 4ª edição. - São Paulo: Pearson Prentice Hall, 2008.

STANGER, James; LANE, Patrick T. Lane. **REDE SEGURA LINUX**. Seu guia de segurança em programas de código aberto. Alta Books, 2002.
SICA, Carlos. **PHP com tudo. Editora:** Ciência Moderna, São Paulo – SP. 2011.

SILVA;VIDEIRA. Alberto Manuel Rodrigues, Carlos Alberto Escaleira. **UML, Metodologias de ferramentas CASE**, Porto Lisboa - Portugal, Centro Atlântico, Ltda., 2001, p. 222.

TORRES, Gabriel. **Redes de computadores –** Curso completo. Rio de Janeiro: Axcel Books 2001, Cap. 1. p. 5.

TANENBAUM, Andrew S. **Redes de computadores** - Tradução da 4ª Edição. 2003. 2003. p. 543.

REHEM, Almerindo; BRANDÃO, Nicole Gonçalves; JUNIOR, Ubirajara de B. Cruz. **Manual de Instalação de Servidor Squid baseado no Linux CENTOS 5.4.** Tiradentes de Aracaju, 2010. <http://www.devin.com.br/?option=com_content&view=article&id=100:manual-do-squid&catid=43:trabalhos-de-alunos&Itemid=86> Acesso em 29/05/2016.

APÊNDICE A – Termo de Autorização de Implantação Colégio Maria Imaculada

centro universitário unifacvest

TERMO DE AUTORIZAÇÃO PARA IMPLANTAÇÃO DE TRABALHO DE CONCLUSÃO DE CURSO II

Centro Universitário Unifacvest
Acadêmico: Rodrigo Ribeiro
Curso: Ciência da Computação - Fase: 8ª
Lages, SC

Local: Colégio Maria Imaculada.
Rua Madre Iva Poupon, 69 - N. Sª. Aparecida - CEP: 89520-000
Curitibanos/SC

Eu Alvacir Merini, após conhecer e entender os objetivos, procedimentos e benefícios deste trabalho, bem como estar ciente da necessidade do uso software e utilização do local especificados no projeto, através do presente termo, autorizo o acadêmico Rodrigo Ribeiro a implantar o Trabalho de conclusão de Curso II projeto de pesquisa intitulado "Firewall Iptables e Squid Web Proxy Shell" a realizar o trabalho, sem quaisquer ônus financeiros a nenhuma das partes. Ao mesmo tempo, libero a utilização do laboratório para alocação de espaço para implantação do servidor COMPUTADOR DELL OPTIPLEX 790 CORE I5 4GB 320GB HD para fins científicos e de estudos (livros, artigos, slides e transparências), em favor do pesquisador, acima especificado.

Curitibanos, 10 de Outubro de 2016.

Alvacir Merini
Diretora
RG: 9.344.382-1
Aut. nº 5860

Rodrigo Ribeiro
Responsável pelo projeto

Assinatura e Carimbo do local

APÊNDICE B – Classe bloquear sites Squid Web Proxy Shell evento bloquear

```
 <!DOCTYPE html>
<body>
  <script Language="JavaScript">
    function Aplicar() {
    window.open( "aplica.php", "Aplicando Regras", "status = 1, height = 250, width =
500, resizable = 20" )
    }
  </script>
  <head>
    <title></title>
    <meta name='viewport' content='width=device-width, initial-scale=1, maximum-
scale=1'>
    <script type="text/javascript" src="http://code.jquery.com/jquery-
1.11.3.min.js"></script>
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css">
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap-theme.min.css">
    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js"></script>
  </head>
  <div class="fullscreen_bg_two">
//evento php inserir sites no arquivo urlbloqueados gerenciado pelo squid.conf
    <?php
    $local = "/etc/squid/urlbloqueados.txt";
    if(count($_POST) > 0)
    {
      $arquivo = fopen("$local", "w+");
      fwrite($arquivo, $_POST['arquivo']);
      fclose($arquivo);
      exec("sudo /usr/bin/sort $local -o $local");
      exec("sudo /usr/sbin/squid -k reconfigure");
    }
    $arquivo = fopen("$local", "r");
    ?>
  </body>
  <style>
   .btn {
    position:absolute;
    top: 300px;
    left: 75;
  }
```

APÊNDICE C – Classe Menu Inicial Squid Web Proxy Shell

```
<!DOCTYPE html>
<head>
  <meta charset="utf-8">
  <link href="css/bootstrap.css" rel="stylesheet">
  <!-- Chamadas JS -->
  <script src="js/jquery.min.js"></script>
  <script src="js/bootstrap.min.js"></script>
  <link rel="stylesheet" href="css/style.css"/>
  <script src="js/bootbox.min.js"></script>
  <link rel="stylesheet" href="bs.css">
 <script src="jquery.js"></script>
 <script src="bs.js"></script>
</head>
<body>
 <div class="fullscreen_bg_two">
   <div class="container">
     <nav class="navbar navbar-inverse navbar-fixed-top">
       <div class="container-fluid">
        <!-- Logo -->
        <div class="navbar-header">
          <a href="index.php" class="navbar-brand ">Squid Web Proxy Shell</a>
           </div>
     <!-- Menu Items -->
     <div>
          <ul class="nav navbar-nav">
        <!-- drop down menu -->
            <li class="dropdown">
              <a href="#" class="dropdown-toggle" data-
toggle="dropdown">RelatÃ³rios<span class="caret"></span>
            </a>
            <ul class="dropdown-menu">
             <li><a href="http://localhost/squid-reports/">SARG</a></li>
              <li><a href="http://localhost:3000">NTOP/a></li>
             </ul>
            </li>
          </ul>
          <ul class="nav navbar-nav">
            <li class="active"></li>
           <li class="dropdown">
              <a href="#" class="dropdown-toggle" data-toggle="dropdown">Gerenciar
Bloqueio<span class="caret"

            ></span></a>

            <ul class="dropdown-menu">
```

```
            <li><a href="bloquear_sites.php">Bloquear - Sites</a></li>
          <li><a href="bloquear_ips.php">Bloquear - IP</a></li>
              <li><a href="bloquear_login.php">Bloquear - Logins</a></li>
              <li><a href="bloquear_palavras.php">Bloquear - Palavras</a></li>
            </ul>
           </li>
          </ul>
          <ul class="nav navbar-nav">
           <li class="active"></li>
           <!-- drop down menu -->
           <li class="dropdown">
            <a href="#" class="dropdown-toggle" data-toggle="dropdown">Gerenciar
Liberacão<span class="caret"
            ></span></a>
            <ul class="dropdown-menu">
              <li><a href="liberar_sites.php">Liberar - Sites</a></li>
              <li><a href="liberar_ip.php">Liberar - IP</a></li>
              <li><a href="liberar_login.php">Liberar - Login</a></li>
              <li><a href="liberar_palavras.php">Liberar - Palavras</a></li>
            </ul>
           </li>
         </ul>
         <!--right align -->
         <ul class="nav navbar-nav navbar-right">
          <li><a href="index.php">Sair</a></li>
        </ul>
      </div>
      <footer>
        <div class="navbar navbar-inverse navbar-fixed-bottom">
          <div class="container">
            <div class="navbar-collapse collapse" id="footer-body">
              <ul class="nav navbar-nav">
               <img src="img/logo.jpg" height="40">
               <img src="img/bicho.png" height="40">
              </ul>
            </div>
            <div class="navbar-header">
             <button type="button" class="navbar-toggle" data-toggle="collapse" data-
target="#footer-body">
              <span class="icon-bar"></span>
              <span class="icon-bar"></span>
              <span class="icon-bar"></span>
            </button>
            <ul class="footer-bar-btns visible-xs">

              <li><a href="#" class="btn" title="History"><i class="fa fa-2x fa-clock-o
blue-text"></i></a></li>
```

```
            <li><a href="#" class="btn" title="Favourites"><i class="fa fa-2x fa-star
yellow-text"></i></a></li>
            <li><a href="#" class="btn" title="Subscriptions"><i class="fa fa-2x fa-rss-
square orange-text">
            </i></a></li>
          </ul>
        </div>
      </div>
    </div>
  </footer>
 </div>
</nav>
</body>
</html>
```

APÊNDICE D – Classe liberar sites Squid Web Proxy Shell

```
<!DOCTYPE html>

<body>

  <script Language="JavaScript">

  function Aplicar() {

  window.open( "aplica.php", "Aplicando Regras", "status = 1, height = 300, width =
250, resizable = 15" )

    }

  </script>

  <head>

    <title></title>

    <meta name='viewport' content='width=device-width, initial-scale=1, maximum-
scale=1'>

    <script type="text/javascript" src="http://code.jquery.com/jquery-
1.11.3.min.js"></script>

    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap.min.css">

    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/css/bootstrap-theme.min.css">

    <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.4/js/bootstrap.min.js"></script>

  </head>

  <?php

  $local = "/etc/squid/liberadoteste.txt";

  if(count($_POST) > 0)

  {
```

```
  $arquivo = fopen("$local", "w+");
  fwrite($arquivo, $_POST['arquivo']);
  fclose($arquivo);
  exec("sudo /usr/bin/sort $local -o $local");
  exec("sudo /usr/sbin/squid -k reconfigure");
}
$arquivo = fopen("$local", "r");
?>
```

APÊNDICE E – Arquivo de configuração FIREWALL

```
#! /bin/bash

#Configuracao Firewall atraves do iptables
#Autoria do Script
#":::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
#"| Script de Firewall - IPTABLES"
#"| Criado por: Rodrigo Ribeiro
#"| guigo_ribeiro@hotmail.com
#"| Uso: firewall start|stop|restart"
#":::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"

############################
#######TITULO ABRE####### #
echo "Iniciando FIREWALL"
############################

#########################################################################
#Os diversos modulos do iptables sao chamados atraves do modprobe#
#########################################################################

modprobe ip_tables
modprobe iptable_nat
modprobe ip_conntrack
modprobe ip_conntrack_ftp
modprobe ip_nat_ftp
modprobe ipt_LOG
modprobe ipt_REJECT
modprobe ipt_MASQUERADE
modprobe ipt_state
modprobe ipt_multiport
modprobe iptable_mangle
modprobe ipt_tos
modprobe ipt_limit
modprobe ipt_mark
modprobe ipt_MARK

echo "subindo os modulos do IPTABLE e NETFILTRE"
echo "ON .....[#OK]"

####################
#Interfaces de Rede#
####################
LAN=eth0

echo "ativado o placa de rede eth0"
echo "ON .....[#OK]"

############################
```

```
#Mensagem de inicializacao#
############################

echo "::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
echo "| Script de Firewall - IPTABLES"
echo "| Criado por: Rodrigo Ribeiro"
echo "| TCC 2 - CENTRO UNIVERSITÃ•  RIO UNIFACVEST"
echo "| guigo_ribeiro@hotmail.com"
echo "| Uso: firewall start|stop|restart"
echo "::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
echo
echo "=========================================================|"
echo "|:INICIANDO A CONFIGURACAO DO FIREWALL NETFILTER ATRAVES:|"
echo "|:                  DO IPTABLES                    :|"
echo "=========================================================|"

######################
#Zera todas as Regras#
######################

echo "Zera todas as Regras"
iptables -F

########################################
##Bloqueia tudo, nada entra e nada sai##
########################################

echo "Fechando tudo..."
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT DROP

#############################################################################
#################################################
# Loga/Adiciona/Descarta hosts da lista "SUSPEITO" (cuja conexÃ£o nÃ£o cumpre nenhuma
das regras acima) {deixe como Ãºltima regra!}
#############################################################################
#################################################
iptables -A INPUT -p tcp --dport=20 -j LOG --log-level warning --log-prefix "[firewall] [ftp]"
iptables -A INPUT -p udp --dport=20 -j LOG --log-level warning --log-prefix "[firewall] [ftp]"
iptables -A INPUT -p tcp --dport=21 -j LOG --log-level warning --log-prefix "[firewall] [ftp]"
iptables -A INPUT -p udp --dport=21 -j LOG --log-level warning --log-prefix "[firewall] [ftp]"
iptables -A INPUT -p tcp --dport=22 -j LOG --log-level warning --log-prefix "[firewall] [ssh]"
iptables -A INPUT -p udp --dport=22 -j LOG --log-level warning --log-prefix "[firewall] [ssh]"
iptables -A INPUT -p tcp --dport=23 -j LOG --log-level warning --log-prefix "[firewall]
[telnet]"
iptables -A INPUT -p udp --dport=23 -j LOG --log-level warning --log-prefix "[firewall]
[telnet]"
iptables -A INPUT -p icmp  -j LOG --log-level warning --log-prefix "[firewall] [ping]"

#################################
```

```
###Libera conexoes estabelecidas###
####################################

iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A FORWARD -m state --state RELATED,ESTABLISHED,NEW -j ACCEPT
iptables -A OUTPUT -m state --state RELATED,ESTABLISHED,NEW -j ACCEPT

iptables -A INPUT -i lo -j ACCEPT

echo "Liberando conexoes..."
echo "ON.....[#ON]"

###################
#COMPARTILHAMENTO##
###################

iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

echo "Libera Compartilhamento"
echo "ON.....[#ON]"

############################################
#Bloqueio de scanners ocultos (Shealt Scan) portscanners, ping of death, ataques DoS, pacotes danificados e etc#
############################################

iptables -A FORWARD -p tcp --tcp-flags SYN,ACK, FIN,  -m limit --limit 1/s -j ACCEPT
iptables -A FORWARD -p icmp --icmp-type echo-request -m limit --limit 1/s -j ACCEPT
iptables -A INPUT -p icmp --icmp-type echo-request -m limit --limit 1/s -j ACCEPT
iptables -A INPUT -i eth0 -p icmp --icmp-type echo-reply -m limit --limit 1/s -j DROP
iptables -A FORWARD -p tcp -m limit --limit 1/s -j ACCEPT
iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A FORWARD -p tcp --tcp-flags SYN,ACK,FIN,RST RST -m limit --limit 1/s -j
ACCEPT
iptables -A FORWARD --protocol tcp --tcp-flags ALL SYN,ACK -j DROP
iptables -A INPUT -m state --state INVALID -j DROP
iptables -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -N VALID_CHECK
iptables -A VALID_CHECK -p tcp --tcp-flags ALL FIN,URG,PSH -j DROP
iptables -A VALID_CHECK -p tcp --tcp-flags ALL SYN,RST,ACK,FIN,URG -j DROP
iptables -A VALID_CHECK -p tcp --tcp-flags ALL ALL -j DROP
iptables -A VALID_CHECK -p tcp --tcp-flags ALL FIN -j DROP
iptables -A VALID_CHECK -p tcp --tcp-flags SYN,RST SYN,RST -j DROP
iptables -A VALID_CHECK -p tcp --tcp-flags SYN,FIN SYN,FIN -j DROP
iptables -A VALID_CHECK -p tcp --tcp-flags ALL NONE -j DROP

echo "bloqueado scanners ocultos"
echo "ON .....[#OK]"

#########################
#Bloqueio Anti-Spoofings#
```

```
#########################

iptables -A INPUT -s 127.0.0.0/8 -i eth0 -j DROP
iptables -A INPUT -s 127.0.0.0/8 -i eth0 -j DROP
iptables -A INPUT -s 192.168.2.0/24 -i eth0 -j DROP
iptables -A INPUT -s 192.168.2.0/24 -i eth0 -j DROP

echo "ativado o bloqueio de tentativa de ataque do tipo Anti-spoofings"
echo "ON .....[#OK]"

#####################################
#Bloqueio de ataque ssh de forca bruta#
#####################################

iptables -N SSH-BRUT-FORCE
iptables -A INPUT -i $LAN -p tcp --dport 22 -j SSH-BRUT-FORCE
iptables -A SSH-BRUT-FORCE -m limit --limit 1/s --limit-burst 4 -j RETURN
iptables -A SSH-BRUT-FORCE -j DROP

echo "ativado o bloqueio a tentativa de ataque do tipo SSH-BRUT-FORCE"
echo "ON .....[#OK]"

#################################
#Bloquear ataque do tipo SYN-FLOOD#
#################################

echo "0" > /proc/sys/net/ipv4/tcp_syncookies
iptables -N syn-flood
iptables -A INPUT -i $LAN -p tcp --syn -j syn-flood
iptables -A syn-flood -m limit --limit 1/s --limit-burst 4 -j RETURN
iptables -A syn-flood -j DROP

echo "ativado o bloqueio de ataque do tipo SYN-FLOOD"
echo "ON .....[#OK]"

################################################
#Descarte de pacotes nao-identificado ICMP (ping)
################################################

iptables -A OUTPUT -m state -p icmp --state INVALID -j DROP

echo "ativado o bloqueio a tentativa de ataque do tipo PING-ICMP"
echo "ON .....[#OK]"

######################
#Contra Pings da morte
######################

echo "0" > /proc/sys/net/ipv4/icmp_echo_ignore_all
iptables -N PING-MORTE
iptables -A INPUT -p icmp --icmp-type echo-request -j PING-MORTE
```

```
iptables -A PING-MORTE -m limit --limit 1/s --limit-burst 4 -j RETURN
iptables -A PING-MORTE -j DROP

echo "ativado o bloqueio a ataque do tipo ping da morte"
echo "ON .....[#OK]"

#############################################################################
##Impede ataques DoS a maquina limitando a quantidade de respostas do ping##
#############################################################################

iptables -A INPUT -p icmp --icmp-type echo-request -m limit --limit 1/s -j DROP

echo "Previne ataques DoS"
echo "ON .....[#OK]"

##################################
##Bloqueia completamente o ping##
##################################

iptables -A INPUT -p icmp --icmp-type echo-request -j DROP

echo "Previne ataques PING"
echo "ON .....[#OK]"

############################
##Politicas de seguranca##
############################

echo 0 > /proc/sys/net/ipv4/conf/all/accept_source_route
echo 0 > /proc/sys/net/ipv4/conf/all/accept_redirects
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
echo 1 > /proc/sys/net/ipv4/tcp_syncookies
echo 1 > /proc/sys/net/ipv4/conf/default/rp_filter
iptables -A INPUT -m state --state INVALID -j DROP

echo "Implementacao de politicas de seguranca"
echo "ON .....[#OK]"

echo "bloqueio facebook OK"

#-----------------------------------------------------------#
iptables -A INPUT -s www.facebook.com -d 192.168.0.102/24 -j DROP
#-----------------------------------------------------------#

#-----------------------------------------------------------#
iptables -A INPUT -s www.twitter.com -d 192.168.0.102/24 -j DROP
#-----------------------------------------------------------#

#-----------------------------------------------------------#
iptables -A INPUT -s www.youtube.com -d 192.168.0.102/24 -j DROP
#-----------------------------------------------------------#
```

```
echo 1 >/proc/sys/net/ipv4/ip_forward

#--------------------------------------------------------------------------------------------------------#
#redirecionamento de tudo que for da rede 192.168.2.0/24 para porta 80 redirecionar para porta (3128) proxy
#--------------------------------------------------------------------------------------------------------#

iptables -t nat -A PREROUTING -s 192.168.2.0/24 -p tcp --dport 80 -j REDIRECT --to-port 3128

#-----------------------------------------------------------------#
#liberando somente a porta 25 para envio de email pelo outlook express
#-----------------------------------------------------------------#

iptables -t nat -A POSTROUTING -s 192.168.2.0/24 -p tcp -o eth0 --dport 25 -j MASQUERADE

#-----------------------------------------------------------------#
#liberando somente a porta 110 para receber email pelo outlook express
#-----------------------------------------------------------------#

iptables -t nat -A POSTROUTING -d 192.168.2.0/24 -p tcp --dport 110 -o eth0 -j MASQUERADE

#-------------------------------------------------------------------------------------------------------------
----#
#libero tudo que for de origem 192.168.2.0/24 para internet posso cancelar a regra que libera a porta 25 para enviar email
#-------------------------------------------------------------------------------------------------------------
----#

iptables -t nat -A POSTROUTING -s 192.168.2.0/24 -o eth0 -j MASQUERADE

#--------------------------------------------------------------------------------------------------------#
#libero tudo para entrar na rede 192.168.2.0/24 posso cancelar a regra que libera a porta 110 para receber
#--------------------------------------------------------------------------------------------------------#

iptables -t nat -A POSTROUTING -d 192.168.2.0/24 -o eth0 -j MASQUERADE

##########################################################################################
##Libera o acesso via SSH e Limita o numero de tentativas de acesso a 4 a cada minutos##
##########################################################################################

iptables -I INPUT -p tcp --dport 22 -i eth0 -m state --state NEW -m recent --update --seconds 60 --hitcount 4 -j DROP
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
iptables -A INPUT -p udp --dport 22 -j ACCEPT
```

```
# [ SSH Interno ]
iptables -t nat -A PREROUTING -p tcp --dport 22 -s 192.168.1.0/24 -j ACCEPT
iptables -t nat -A PREROUTING -p udp --dport 22 -s 192.168.1.0/24 -j ACCEPT
iptables -A INPUT -p tcp --dport 22 -s 192.168.2.0/24 -j ACCEPT
iptables -A INPUT -p udp --dport 22 -s 192.168.2.0/24 -j ACCEPT

# [ SSH Externo ]
iptables -t nat -A PREROUTING -p tcp --dport 22 -j ACCEPT
iptables -t nat -A PREROUTING -p udp --dport 22 -j ACCEPT
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
iptables -A INPUT -p udp --dport 22 -j ACCEPT

echo "Liberando SSH"
echo "On.....[#OK]"

###################
##Libera o Samba##
###################

iptables -A INPUT -p tcp --dport 137:139 -j ACCEPT
iptables -A INPUT -p udp --dport 137:139 -j ACCEPT
iptables -A INPUT -p tcp --dport 445 -j ACCEPT
iptables -A INPUT -p icmp -j ACCEPT
iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -p udp --dport 137 -j ACCEPT
iptables -A INPUT -p udp --dport 138 -j ACCEPT
iptables -A INPUT -p tcp --dport 139 -j ACCEPT
iptables -A INPUT -p tcp -j REJECT --reject-with tcp-reset
iptables -A INPUT -p udp -j REJECT --reject-with icmp-port-unreachable
iptables -A INPUT -j REJECT --reject-with icmp-proto-unreachabl

echo "Liberando Samba Compartilhamento do Samba"
echo "On.....[#OK]"

###################
##Libera o Apache##
###################

echo "Liberando o Apache"
iptables -A INPUT -p tcp --dport 80 -j ACCEPT

########################
#Bloquear Back Orifice#
########################

iptables -A INPUT -p tcp --dport 31337 -j DROP
iptables -A INPUT -p udp --dport 31337 -j DROP

echo "ativado o bloqueio a tentativa de ataque do tipo BACK-ORIFICE"
```

```
echo "ON .....[#OK]"

#################
#Bloquear NetBus#
#################

iptables -A INPUT -p tcp --dport 12345:12346 -j DROP
iptables -A INPUT -p udp --dport 12345:12346 -j DROP

echo "ativado o bloqueio a tentativa de ataque do tipo NET-BUS"
echo "ON .....[#OK]"

################
##TITULO FECHA##
################

echo "Configuracao Firewall Concluida."
echo
echo "=========================================================|"
echo "::TERMINADA A CONFIGURACAO FIREWALL NETFILTER ATRAVES::|"
echo "::                DO IPTABLES                    ::|"
echo "=========================================================|"
echo "FIREWALL ATIVADO - SISTEMA PREPARADO"
echo "SCRIPT DE FIREWALL CRIADO POR :-) RODRIGO RIBEIRO :-)"
echo "FIREWALL DESCARREGADO - SISTEMA LIBERADO"
```

APÊNDICE F – ARQUIVO DE CONFIGURAÇÃO SQUID.CONF

```
#------------------------------------------------------#
#PORTA NA QUAL O SERVIDOR VAI FICAR DISPONÃ•  VEL PORTA 3128
#------------------------------------------------------#

http_port 3128

#NOME DO SERVIDOR

visible_hostname Rodrigo

#------------------------------------------------------#
#CONTROLE DE CACHE DA MEMÃ“RIA
#------------------------------------------------------#

hierarchy_stoplist cgi-bin ?

cache_mem 16 MB

cache_swap_low 90

cache_swap_high 95

maximum_object_size 4096 KB

#------------------------------------------------------#
#ACL QUE PERMITE TODOS OS IPS PARA UTILIZAÃ‡Ãƒ O DO PROXY
#------------------------------------------------------#

acl all src 0/0

acl SSL_ports port 443 #https

acl Safe_ports port 80 # http

acl Safe_ports port 21 # ftp

acl Safe_ports port 443 # https

acl Safe_ports port 70 # gopher

acl Safe_ports port 210 # wais

acl Safe_ports port 1025-65535 # unregistered ports
```

```
acl Safe_ports port 280 # http-mgmt

acl Safe_ports port 488 # gss-http

acl Safe_ports port 591 # filemaker

acl Safe_ports port 777 # multiling http

acl CONNECT method CONNECT

#----------------------------#
#LOG PERSONALIZADO DA GERENCIA
#----------------------------#

logformat MEU_LOG IP do cliente: %>a - Username: %un - Horario: [%tl] - Metodo:
%rm - URL: %ru - Status HTTP: %Hs - Status Squid: %Ss
#ARQUIVO DE GERAÃ‡Ãƒ O DE LOG PADRÃƒO DO SQUID

access_log /var/log/squid/access.log

#ARQUIVO DE GERAÃ‡Ãƒ O DE LOG PERSONALIZADO
access_log /var/log/squid/gerencia.log MEU_LOG

#cache_dir ufs /var/cache/squid 5000 16 256

#REPASSA AS INFORMAÃ‡Ã•ES PARA O PROXY MÃƒE "192.168.80.2"
#proxy-only default
#cache_peer 192.168.80.2 parent 3128 0 no-query no-digest
#SUBMISSÃƒO DE OUTRO SERVIDOR
#never_direct allow all

#----------------------------------------#
# CONFIGURAÃ‡Ã•ES PARA AUTENTICAÃ‡Ãƒ O DO PROXY
#----------------------------------------#

auth_param basic realm squid
auth_param basic program /usr/lib/squid3/basic_ncsa_auth etc/squid/passwd
acl autenticados proxy_auth REQUIRED

#----------------------------------------#
# USUÃ•  RIOS COM ACESSO LIVRE NO PROXY
#----------------------------------------#

#DA ACESSO TOTAL AOS USUARIOS NO ARQUIVO "LIVRE.TXT"
acl acesso_livre proxy_auth "/etc/squid/livre.txt"
http_access allow acesso_livre
http_access allow autenticados acesso_livre
```

```
#---------------------------------------#
# ACL LIBERAÃ‡ÃƒO DE SITES
#---------------------------------------#

acl acl_libera url_regex -i "/etc/squid/liberadoteste.txt"
http_access allow acl_libera
#minha ediÃ§Ã£o

#---------------------------------------#
# ACL PALAVRAS BLOQUEADAS
#---------------------------------------#

acl palavrasproibidas url_regex -i "/etc/squid/palavrasproibidas.txt"
http_access deny palavrasproibidas

#-------------------------------------------------------------------#
# ACL USUÃ□RIOS RESTRITOS SERÃƒO BARRADOS NO ARQUIVO ->
urlbloqueados.txt
#-------------------------------------------------------------------#

#DA ACESSO AOS USUARIOS DO ARQUIVO "RESTRITO.TXT" A TODOS OS
SITES MENOS OS SITES DESCRITOS NO "URLBLOQUEADO.TXT"
acl acesso_restrito proxy_auth "/etc/squid/restrito.txt"
acl url_bloqueado url_regex -i "/etc/squid/urlbloqueados.txt"
http_access deny url_bloqueado
http_access allow acesso_restrito !url_bloqueado
http_access allow autenticados acesso_restrito


#-----------------------------------------------------------------------------------#
#BLOQUEIA O ACESSO DOS DEMAIS USUARIOS A TODOS OS SITES
DESCRITOS NO ARQUIVO "BLOQUEADOS.TXT"
#-----------------------------------------------------------------------------------#

acl bloqueados dstdomain url_regex -i "/etc/squid/bloqueados.txt
http_access deny bloqueados
#BLOQUEIA O ACESSO DOS DEMAIS USUARIOS A TODAS AS PALAVRAS
DESCRITAS NO ARQUIVO "PALAVRASPROIBIDAS.TXT"


#-------------------------------------------------------------------#
#BLOQUEIA O ACESSO DOS DEMAIS USUARIOS A TODOS OS FORMATOS
DESCRITOS
#-------------------------------------------------------------------#

acl extban url_regex -i \.mp3 \.mp4 \.avi
```

```
http_access deny extban

#------------------------------------------------------------------#
#BLOQUEIO DE IPS
#------------------------------------------------------------------#

acl ips_bloq src "/etc/squid/ips.txt"
http_access deny ips_bloq

#------------------------------------------------------------------#
#LIBERAÃ‡ÃƒO DE IPS
#------------------------------------------------------------------#

acl ips_bloq src "/etc/squid/ips.txt"
http_access allow ips_bloq

#------------------------------------------------------------------------------------------------------------
-------------------------#
#BLOQUEIA O ACESSO DOS DEMAIS USUARIOS A TODOS OS SITES EM
DETERMINADO DIAS DA SEMANA "SMTWHFA - COMEÃ‡ANDO DE S-
DOMINGO ATÃ‰ A-SÃ•  BADO"
#------------------------------------------------------------------------------------------------------------
-------------------------#

#acl horario time SMTWHFA 16:30-18:10

#http_access deny horario

#acl descansar time 03:00-07:00

#http_access deny desacansar

#DANDO PERMISSÃƒO A ACL "ALL"

http_access allow all
```