UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

COORDENAÇÃO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

HENDRIKUS FRANCISCO RESENDE

WAGNER CORRÊA DE OLIVEIRA STELLA

PROXY TRANSPARENTE APLICADO EM UM SERVIDOR INSTITUCIONAL

TRABALHO DE CONCLUSÃO DE CURSO

PONTA GROSSA

2015

HENDRIKUS FRANCISCO RESENDE

WAGNER CORRÊA DE OLIVEIRA STELLA

PROXY TRANSPARENTE APLICADO EM UM SERVIDOR INSTITUCIONAL

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Ms. Rogério Ranthum

PONTA GROSSA

2015

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Ponta Grossa

Diretoria de Graduação e Educação Profissional
Coordenação de Análise e Desenvolvimento de Sistemas
Tecnologia em Análise e Desenvolvimento de Sistemas

TERMO DE APROVAÇÃO

PROXY TRANSPARENTE APLICADO EM UM SERVIDOR INSTITUCIONAL

por

HENDRIKUS FRANCISCO RESENDE

WAGNER CORRÊA DE OLIVEIRA STELLA

Este(a) Trabalho de Conclusão de Curso foi apresentado(a) em 22 de maio de 2015 como requisito parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas. Os candidatos foram arguidos pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

_____
Prof. Msc. Rogério Ranthum
Prof. Orientador

_____
Prof. Dr. Geraldo Ranthum
Membro titular

_____
Prof. Dr. Richard Duarte Ribeiro
Membro titular

- O Termo de Aprovação assinado encontra-se na Coordenação do Curso

A Deus, que se mostrou criador, que foi criativo.
Seu fôlego de vida em nós e que nos foi sustento e nos deu coragem para questionar realidades e propor sempre um novo mundo de possibilidades.

AGRADECIMENTOS

Em primeiro lugar dedicamos o nosso trabalho a Deus.

A nossas famílias que nos apoiaram em toda nossa trajetória pessoal e acadêmica.

A nosso orientador Professor Mestre Rogério Ranthum.

Ao corpo docente, pedagógico e administrativo da Universidade Tecnológica Federal do Paraná - Campus Ponta Grossa.

E a todos que de alguma maneira nos ajudaram a chegar até aqui.

“A ciência de hoje é a tecnologia de amanhã.”
(TELLER, Edward)

RESUMO

RESENDE, Hendrikus Francisco. STELLA, Wagner Côrrea de Oliveira. ***Proxy* Transparente Aplicado em um Servidor Institucional.** 2015. Setenta e sete folhas. Trabalho de Conclusão de Curso. Tecnologia em Análise e Desenvolvimento de Sistemas - Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2015.

Pensando no ambiente da Universidade Tecnológica Federal do Paraná - Campus Ponta Grossa, que conta com diferentes redes locais e diferentes configurações necessárias para se obter acesso a internet, mantendo os padrões de segurança da instituição, optou-se por simular o ambiente heterogêneo do campus e nele implementar um servidor *Proxy* de maneira transparente, para que não seja mais necessário que o usuário precise saber uma faixa de *IP'S* ou necessite se autenticar para que consiga fazer uso da internet do local. O servidor, microcomputador de arquitetura x86, com duas placas rede, recebe uma rede externa com conexão à internet, pela placa de rede denominada eth0, essa placa faz o roteamento de internet para outra placa, denominada eth1. A partir desse processo temos duas redes com faixas de *IP'S* diferentes, uma para alunos e outra para os setores administrativo/pedagógico. Para garantir a portabilidade do serviço proposto, a conexão com a rede foi testada em diversos sistemas operacionais como Linux, Windows e Android. Para garantir que todos os navegadores estivessem requisitando e recebendo pacotes de dados, foram testados nos navegadores mais populares como Internet Explorer, Mozilla Firefox e Google Chrome. Com todos os testes realizados, obteu-se um resultado positivo, garantindo com que o serviço proposto seja uma alternativa viável para execução.

Palavras-chave: Redes de Computadores. Segurança. Servidor *Proxy*. *Proxy* Transparente

ABSTRACT

RESENDE, Hendrikus Francisco. STELLA, Wagner Côrrea de Oliveira. **Transparent *Proxy* applied in an Institutional Server.** 2015. Seventy-seven leaves. Conclusion Work. Technology Analysis and Systems Development - Federal Technology University - Parana. Ponta Grossa, 2015.

Thinking about environment of the University of Paraná Federal Technological - Campus Ponta Grossa, which has different locations and different network configurations needed to get internet access while maintaining the institution's safety standards, it was decided to simulate the heterogeneous campus environment Implement a *Proxy* server transparently, so that you no longer need the user having to know one ips *RANGE* or need to authenticate so you can make use of the internet site. The server architecture of *PC* x86, with two network cards, receives an external network with Internet connection, the network card called eth0, this card does internet routing to other card, called eth1. From this process we have two networks with different *IP* ranges, one for students and one for administrative / educational sectors. To ensure portability of the proposed service, the connection to the network was tested in several operating systems like Linux, Windows and Android. To ensure that all browsers were requesting and receiving data packets, they were tested in the most popular browsers like Internet Explorer, Mozilla Firefox and Google Chrome. With all tests, are a positive outcome, ensuring that the proposed service is a viable alternative to execution.

Keywords: Computer Networks. Security. *Proxy* Server. *Proxy* Transparent.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE SIGLAS E ACRÔNIMOS

ACL	Access List
AES	Advanced Encryption Algorithm
AP	Access Point
DES	Data Encryption Standard
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DSS	Decision Support Systems
FTP	File Transfer Protocol
HTTP	Hyper Text Transfer Protocol
HTTPS	Hyper Text Transfer Protocol Secure
IEEE	Institute of Electric and Electronic Engineers
IP	Internet Protocol
ISM	Industrial, Scientific and Medical
LAN	Local Area Netowork
LDAP	Lightweight Directory Access Protocol
MAN	Metropolitan Area Network
MBPS	Megabit por Segundo
MCT	Ministério da Ciência e Tecnologia
NIST	National Institute of Standards and Technology
RC4	ARCFOUR
RSA	Rivest, Shamir e Adleman
SNMP	Simple Network Management Protocol
SSL	Secure Socket Layers

TLS	Transport Layer Security
WAN	Wide Area Network
WEP	Wired Equivalent Privacy
WLAN	Wireless Local Area Network
WPA	Wi-Fi Protected Access

SUMÁRIO

1 INTRODUÇÃO

O *case* apresentado consiste na simulação de como seria o processo de implementação e a execução de um *Proxy* transparente que abranja todos os níveis de rede existentes hoje num ambiente extremamente diversificado como é o da Universidade Tecnológica Federal do Paraná – Câmpus Ponta Grossa.

Como partes do embasamento teórico, são apresentados conceitos de extrema importância para execução do projeto em sua totalidade. Conhecimentos como definições de rede e suas infra-estruturas, modalidades, padronizações e a gerência de redes. Após, foram apresentados conceitos de segurança, já que são interinamente ligados ao assunto de redes, e suas formas de aplicação. Por fim foi apresentado o histórico da instituição para que haja um entendimento macro do ambiente simulado. Na sequência é apresentada a execução do projeto. Desde a configuração básica do servidor levantado, dos equipamentos que compõe a tipologia apresentada, configurações de script nos diversos serviços executados, integração de interfaces de rede, *firewall*, *Proxy* e por fim testes realizados em alguns sistemas operacionais, compatibilidade em navegadores, bem como a comparação entre as redes vigentes na instituição com as simuladas.

Foi necessário um estudo do escopo da rede da UTFPR, bem como os perfis de usuários que a acessam, para que se pudesse manter a fidedignidade da simulação por parte do projeto. Com a necessidade de que o ambiente estudado se apresente mais prático para todos que a utilizam, sem a necessidade de autenticação de usuário, chegou-se a conclusão que um *Proxy* transparente resolveria a situação.

O *Proxy* por si só garante que requisições feitas à Internet sejam recebidas de maneira segura. A proposição de que ele seja implementado de forma transparente surge da necessidade de que se mantenha esse serviço de maneira que os usuários não necessitem conhecer sua existência e muito menos precisem memorizar endereços *IP* para que possam ter acesso a seus conteúdos em seus mais variados níveis de acesso, respeitando as diretrizes de seguranças impostas.

2 JUSTIFICATIVA

Em um ambiente heterogêneo como o da Universidade Tecnológica Federal do Paraná, tem-se a necessidade de acesso a informação nos mais variados níveis, como: setores acadêmico, pedagógico e administrativo, criando desta maneira uma série de dificuldades na administração dessa diversidade de ambientes. Há a necessidade de implementação de uma política de segurança que atenda a toda a demanda de serviços e flexibilizando ao máximo o uso dos recursos disponibilizados pela internet, dada a insatisfação dos servidores públicos da instituição em terem que fazer uma nova configuração em seu navegador a cada mudança de rede dentro da unidade, considerando que existem várias redes com diferentes níveis de acesso, devido a essa gama de níveis de acesso temos como objetivo facilitar a usabilidade e a confiabilidade no serviço que já está em funcionamento e aprimorar para que possa ser de satisfação total dos usuários sem causar transtornos como é a configuração do *Proxy* em determinados momentos.

3 OBJETIVO

3.1 OBJETIVO GERAL

Implementar o uso do *Proxy* Transparente em um ambiente heterogêneo tanto de computação como do tipo de usuários e suas permissões de acesso.

3.2 OBJETIVO ESPECIFICO

- Análise do ambiente;
- Coleta de informações;
- Especificar serviço a ser utilizado;
- Abordar conceitos de redes e assuntos relacionados ao assunto de *Proxy*;
- Desenvolver scripts e implementar o *Proxy* no servidor institucional;
- Realizar testes de segurança e aceitabilidade do que foi executado;
- Descrever os resultados alcançados pelos testes.

4 HISTÓRICO DA INSTITUIÇÃO

Segundo o portal da UTFPR (2014) A história da Universidade Tecnológica Federal do Paraná – UTFPR teve início no século passado. Sua trajetória começou com a criação das Escolas de Aprendizes Artífices em várias capitais do país pelo então presidente, Nilo Peçanha, em 23 de setembro de 1909. No Paraná, a escola foi inaugurada no dia 16 de janeiro de 1910, em um prédio da Praça Carlos Gomes.

O ensino era destinado a garotos de camadas menos favorecidas da sociedade, chamados de “desprovidos da sorte”. Pela manhã, esses meninos recebiam conhecimentos elementares (primário) e, à tarde, aprendiam ofícios nas áreas de alfaiataria, sapataria, marcenaria e serralheria. Inicialmente, havia 45 alunos matriculados na escola, que, logo em seguida, instalou seções de Pintura Decorativa e Escultura Ornamental.

Aos poucos, a escola cresceu e o número estudantes aumentou, fazendo com que se procurasse uma sede maior. Então, em 1936, a Instituição foi transferida para a Avenida Sete de Setembro com a Rua Desembargador Westphalen, onde permanece até hoje. O ensino tornou-se cada vez mais profissional até que, no ano seguinte (1937), a escola começou a ministrar o ensino de 1º grau, sendo denominada Liceu Industrial do Paraná.

Cinco anos depois (1942), a organização do ensino industrial foi realizada em todo o país. A partir disso, o ensino passou a ser ministrado em dois ciclos. No primeiro, havia o ensino industrial básico, o de mestria e o artesanal. No segundo, o técnico e o pedagógico. Com a reforma, foi instituída a rede federal de instituições de ensino industrial e o Liceu passou a chamar-se Escola Técnica de Curitiba. Em 1943, tiveram início os primeiros cursos técnicos: Construção de Máquinas e Motores, Edificações, Desenho Técnico e Decoração de Interiores.

Antes dividido em ramos diferentes, em 1959 o ensino técnico no Brasil foi unificado pela legislação. A escola ganhou, assim, maior autonomia e passou a chamar-se Escola Técnica Federal do Paraná. Em 1974, foram implantados os primeiros cursos de curta duração de Engenharia de Operação (Construção Civil e Elétrica).

Quatro anos depois (1978), a Instituição foi transformada em Centro

Federal de Educação Tecnológica do Paraná (Cefet-PR), passando a ministrar cursos de graduação plena. A partir da implantação dos cursos superiores, deu-se início ao processo de “maioridade” da Instituição, que avançaria, nas décadas de 80 e 90, com a criação dos Programas de Pós-Graduação.

Em 1990, o Programa de Expansão e Melhoria do Ensino Técnico fez com que o Cefet-PR se expandisse para o interior do Paraná, onde implantou unidades. Com a Lei de Diretrizes e Bases da Educação, de 1996, que não permitia mais a oferta dos cursos técnicos integrados, a Instituição, tradicional na oferta desses cursos, decidiu implantar o Ensino Médio e cursos de Tecnologia. Em 1998, em virtude das legislações complementares à LDBE, a diretoria do então Cefet-PR tomou uma decisão ainda mais ousada: criou um projeto de transformação da Instituição em Universidade Tecnológica.

Após sete anos de preparo e o aval do governo federal, o projeto tornou-se lei no dia 7 de outubro de 2005. O Cefet-PR, então, passou a ser a UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ (UTFPR) – a primeira especializada do Brasil. Atualmente, a Universidade Tecnológica conta com 13 câmpus, distribuídos nas cidades de Apucarana, Campo Mourão, Cornélio Procópio, Curitiba, Dois Vizinhos, Francisco Beltrão, Guarapuava, Londrina, Medianeira, Pato Branco, Ponta Grossa, Santa Helena e Toledo.

Das diferentes denominações à primeira Universidade Tecnológica do Brasil:

- 1909 – Escola de Aprendizes Artífices do Paraná
- 1937 – Liceu Industrial do Paraná
- 1942 – Escola Técnica de Curitiba
- 1959 – Escola Técnica Federal do Paraná
- 1978 – Centro Federal de Educação Tecnológica do Paraná – Cefet-PR
- 2005 – Universidade Tecnológica Federal do Paraná – UTFPR

5 REFERENCIAL TEÓRICO

Este capítulo consiste em realizar uma revisão dos trabalhos já existentes sobre o tema abordado, para que assim tenhamos uma base bem fundamentada para tratarmos do caso de uso posteriormente.

5.1 INTRODUÇÃO A REDES DE COMPUTADORES

Segundo Torres (2001), mesmo fora do ambiente da informática, todos têm contato com algum tipo de rede em maior ou menor proporção. Em um supermercado cada caixa registradora pode ser um computador, que, além de estar somando o total a ser pago, está instantaneamente diminuindo a quantidade do produto no estoque do supermercado. O funcionário responsável pelo estoque tem acesso em tempo real, à lista exata de mercadorias que tem dentro do supermercado, assim como o responsável pelas finanças tem acesso ao fluxo de caixa daquele momento, facilitando enormemente o processo de gerência, controle e logística do supermercado.

Caixas eletrônicos de bancos são exemplos de uso diário de redes. Cada terminal é composto estruturalmente de um computador ligado a um servidor, computador este que armazena as informações de conta e o histórico das transações efetuadas. Podem ser extensamente enumerados outros lugares que utilizam redes para comunicação, assim constata-se que as redes surgiram da necessidade da troca de informações de maneira ágil e precisa. Na internet, então, essa troca de informações armazenadas remotamente é levada ao extremo: Dados são armazenados nos locais mais remotos e, na maioria das vezes, o local onde os dados estão fisicamente armazenados não tem a menor importância (Torres, 2001).

O acesso aos dados ocorre em servidores onde na maioria das vezes os usuários não fazem ideia onde eles estejam localizados fisicamente, o que na verdade não importa muito já que queremos poder utilizar, atualizar, excluir os dados aos quais buscamos.

Além da troca de dados, há a vantagem do compartilhamento de periféricos, como impressoras, por exemplo, trazendo assim uma redução de

custos na aquisição de equipamentos. O compartilhamento de recursos, independentemente da localização física do recurso e do usuário é exemplificado num grupo de funcionários de escritório que compartilham uma impressora em comum. Nenhum dos indivíduos realmente necessita de uma impressora privativa, e uma impressora de grande capacidade conectada em rede muitas vezes é mais econômica, mais rápida e de mais fácil manutenção que um grande conjunto de impressoras individuais.

Empresas de grande e médio porte, além de muitas pequenas, têm uma dependência vital de informações em tempo real. Genericamente empresas contam com registros de clientes, estoques, contas a receber, extratos financeiros, informações sobre impostos e muitas outras informações on-line, mesmo com funcionários estando dispersos por dezenas de escritórios ou fábricas em diversos países.

Para Tanembaum (2003), os benefícios não se encontram apenas em aplicações empresariais e comerciais, mas também em aplicações domésticas. Os principais usos hoje registrados são o acesso a informações remotas, comunicação instantânea, entretenimento e comercio eletrônico. O acesso a informações remotas tem várias formas. Ele pode significar navegar na *World Wide Web* para obter informações ou apenas por diversão. A gama de informações disposta na rede mundial de computadores é praticamente infinita. O compartilhamento de recursos nas residências tem grande utilidade tanto quanto em empresas pelos mesmos motivos, porém com menores proporções.

Não necessariamente a rede entre computadores precisa ter estrutura cabeada, dada a mobilidade exigida na execução de algumas atividades, existem as redes sem fio, ou wireless. As redes sem fios têm muitas utilidades, talvez a maior delas esteja no uso de equipamentos eletrônicos portáteis, seja para enviar e receber ligações telefônicas, correio eletrônico, navegar pela *Web*, acessar arquivos remotos e se conectar a máquinas distantes. Além do mais, elas querem fazer isso enquanto se encontram em qualquer lugar do planeta. Um grande exemplo da utilidade wireless são as *web* conferências, os organizadores muitas vezes configuram uma rede sem fio na área de conferência. Qualquer pessoa com um notebook e um modem sem fio pode simplesmente ligar o computador e se conectar a Internet, como se o computador estivesse ligado a uma rede de

fiação. Também muito comum nas universidades que disponibilizam sinal de redes sem fios no campus, para que os alunos possam independente da local, dentro do campus, consultar o acervo da biblioteca, ter informações acadêmicas, ter acesso a pesquisas cientificas ou ler seu correio eletrônico.

Existem alguns tipos de redes, vamos abordar alguns que serão pertinentes.

5.1.1 Tipos de Rede

Para Torres (2001), existem dois tipos básicos de rede: Cliente-Servidor e ponta-a-ponto.

5.1.1.1 Cliente-servidor

No modelo cliente-servidor existem dois processos envolvidos, um na máquina cliente e outro no servidor. A comunicação toma a forma do processo cliente enviando uma mensagem pela rede ao processo servidor. Então, o processo cliente espera por uma mensagem em resposta. Quando o processo servidor recebe a solicitação, ele executa o trabalho solicitado ou procura pelos dados solicitados e envia de volta uma resposta.

O servidor é uma máquina especializada e disponibilizada para realizar apenas essa tarefa, não sendo recomendada a sua utilização contínua. O servidor dedicado responde com mais precisão e agilidade as requisições porque além de ser especializado para realizar a tarefa, normalmente não executa outras tarefas ao mesmo tempo. Hoje no mercado existem diversas opções para o servidor, que pode ou não ser uma máquina instalada fisicamente no local da rede, podendo ser um servidor alocado em outro local acessado diretamente pela internet.

Existem vários tipos de servidores, sendo eles:

- Servidor de Arquivos: Responsável pelo armazenamento de dados - arquivos de texto, planilhas, fotos, vídeos - tem como finalidade compartilhar esses arquivos com usuários dependendo das permissões de cada usuário;
- Servidor de Dados: Basicamente nele é instalado o banco de dados,

simultaneamente os dados podem ser acessados por vários usuários, fazendo com que todos os dados fiquem sincronizados;

● Servidor de Impressão: responsável por processar os pedidos de impressão na rede, classificando por ordem de prioridade;

O modelo cliente-servidor possui algumas características:

● Mínimo de 10 máquinas na rede ou com a necessidade de um grau de segurança mais alto.

● Para implantação é necessário um estudo das utilidades e custos necessários, indicado consultar especialistas.

● Custo maior do que as redes ponto-a-ponto.

● Alta segurança.

● Manutenção especializada, efetuada pelo administrador de rede.

● Máquina Servidora especializada.

●Maior facilidade para evoluções na rede, contudo é necessário o estudo detalhado para que sejam avaliadas as possibilidades e possíveis alterações.

Basicamente o modelo cliente-servidor envolve solicitações e respostas. Amplamente utilizado tanto em redes pequenas quanto em grandes redes.

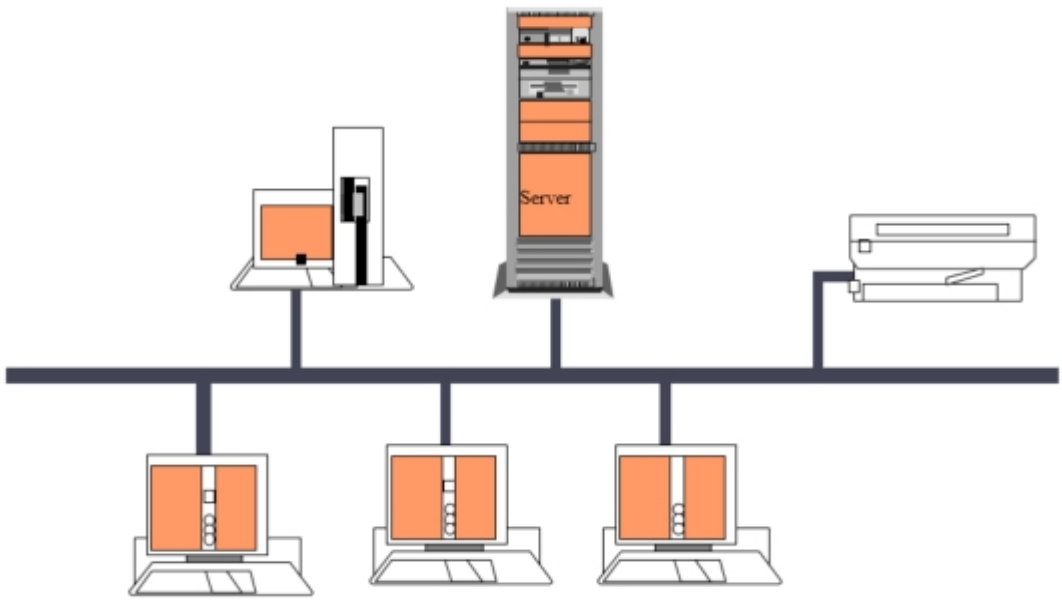

Figura 1 - Rede Cliente-Servidor

Fonte: Autoria própria.

5.1.1.2 Ponto-a-ponto

O modelo ponto-a-ponto, também conhecido como sistema não hierárquico, os micros compartilham os recursos sem muita dificuldade, qualquer micro pode ler e escrever arquivos armazenados em outros micros da rede bem como usar os periféricos que estejam instalados. Sem que haja a máquina servidor essa configuração é efetuada máquina por máquina. Algumas características das redes ponto-a-ponto:

- Baixo custo;
- Fácil implantação;
- Fácil manutenção;
- Baixa segurança;
- Não existe um servidor, cada micro atua como administrador dos seus dados liberando ou não acesso aos demais usuários;
- Posteriormente se necessário a rede terá problemas para aumentar, por isso deve ser feito o planejamento inicial visando mudanças futuras;

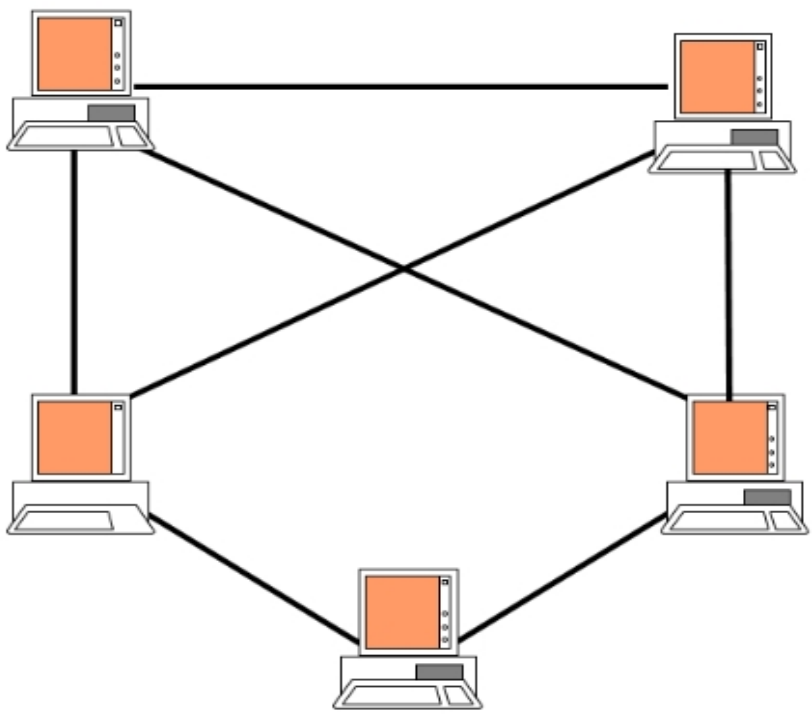

Figura 2 - Rede Ponto-a-ponto
Fonte: Autoria própria.

5.1.2 *Lan*

As redes locais, ou *LAN*, são redes privadas contidas em um único edifício ou ambiente com até alguns quilômetros de extensão. Elas são amplamente usadas para conectar computadores pessoais e estações de trabalho em escritórios e instalações industriais de empresas, permitindo o compartilhamento de recursos, como citado anteriormente, e a troca de informações. As LANs têm três características que as distinguem de outros tipos de redes: (1) tamanho, (2) tecnologia de transmissão e (3) topologia.

As LANs têm um tamanho restrito, o que significa que o pior tempo de transmissão é limitado e conhecido com antecedência. O conhecimento desse limite permite a utilização de determinados tipos de projetos que em outras circunstâncias não seriam possíveis, além de simplificar o gerenciamento da rede.

5.1.3 *Man*

Uma rede metropolitana, ou *MAN*, abrange uma cidade. O exemplo mais conhecido de uma *MAN* é a rede de televisão a cabo disponível em muitas cidades. Esse sistema cresceu a partir de antigos sistemas de antenas comunitárias usadas em áreas com fraca recepção do sinal de televisão pelo ar. Nesses primeiros sistemas, uma grande antena era colocada no alto de colina próxima e o sinal era então conduzido até a casa dos assinantes.

5.1.4 *Wan*

Uma rede geograficamente distribuída, ou *WAN*, abrange uma grande área geográfica, com frequência um país ou continente. Ela contém um conjunto de máquinas cuja finalidade é executar os programas (ou seja, as aplicações) do usuário.

Abaixo uma figura que ilustra uma rede *LAN*, *MAN* e *WAN*:

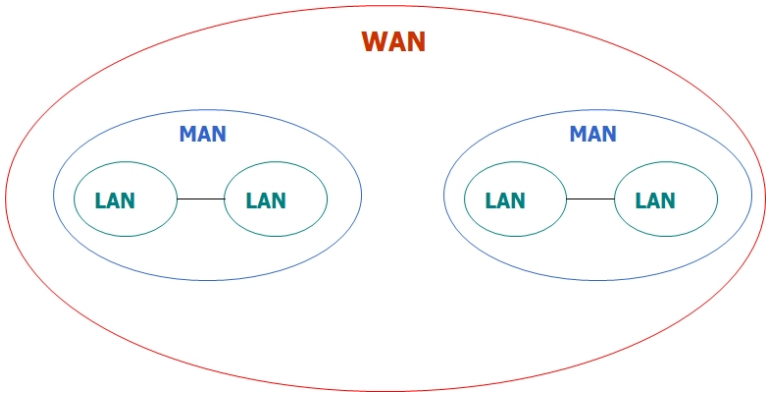

Figura 3 - Rede LAN, MAN e WAN
Fonte: Autoria própria.

5.1.5 *Lan's* e *Wan's* Sem Fio

A comunicação digital sem fios não é uma idéia nova. Em 1901, o físico italiano Guglielmo Marconi demonstrou como funcionava um telégrafo sem fio que transmitia informações de um navio para o litoral por meio de código Morse (afinal de contas, os pontos e traços são binários). Os modernos sistemas digitais sem fios têm um desempenho melhor, mas a idéia básica é a mesma.

Em uma primeira aproximação, redes sem fios podem ser divididas em três categorias principais:

1. Interconexão de sistemas.
2. LANs sem fios.
3. WANs sem fios.

A interconexão de sistemas significa interconectar os componentes de um computador usando rádio de alcance limitado. Quase todo computador tem um monitor, um teclado, um mouse e uma impressora, conectados por cabos à unidade principal. É tão grande o número de novos usuários que enfrentam grande dificuldade para conectar todos os cabos aos pequenos orifícios corretos, mesmo eles sendo codificados com cores que a maioria dos fabricantes de computadores oferece a opção de enviar um técnico à casa do usuário para fazê-lo.

Consequentemente, algumas empresas se uniram para projetar uma rede sem fio de alcance limitado chamada *Bluetooth*, a fim de conectar esses componentes sem a utilização de fios. A rede *Bluetooth* também permite a conexão de câmeras digitais, fones de ouvido, scanners e outros dispositivos a um computador, simplesmente trazendo-os para dentro do alcance da rede. Nada de cabos, nada de instalação de drivers, basta juntá-los, ligá-los e eles funcionarão. Para muitas pessoas, essa facilidade de operação é uma grande vantagem.

Figura 4 - Rede *LAN* e *WAN* SEM FIO
Fonte: Google Imagens = Rede *LAN* e *WAN* sem fio

5.2 GERÊNCIA DE REDES DE COMPUTADORES

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

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

rede;

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

c) Gerência de desempenho – é responsável pela monitoração de desempenho, sua análise e pelo planejamento de capacidade. A monitoração e análise de desempenho baseiam-se basicamente em indicadores, como tempo de resposta, latência da rede, disponibilidade, taxa de erros, entre outros. O planejamento de capacidade vai basicamente demonstrar dados que sugerem a alteração no modo de operação das redes;

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

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

5.3 SEGURANÇA EM INFORMÁTICA

O primeiro passo para se ter segurança na informática é o fator físico que para Torres (2010), é colocar as maquinas em um lugar seguro, longe de condições climáticas e ambientais que possam danificá-la, reduzir a vida útil ou provocar problemas de interrupção do funcionamento a longo prazo, o que também é visto e confirmado em Santo (2010) onde as ameaças à segurança podem ser de diferentes formas como incêndios, inundações, falhas de energia, sabotagem, vandalismo, roubo, e outros. Além do fator físico há a questão de segurança e

integridade das informações contidas no *hardware*, com isso Santo (2010) afirma que o uso da Internet nas organizações trouxe novas vulnerabilidades na rede interna. Se não bastassem as preocupações existentes com espionagem comercial, fraudes, erros e acidentes, agora as empresas também precisam se preocupar com os *hackers*, invasões, vírus e outras ameaças que penetram através desta nova porta de acesso, segundo Tanembaum (2003) problemas com a segurança de conteúdo podem ser divididos nas seguintes áreas interligadas: sigilo, autenticação, não repúdio e controle de integridade. Os sistemas de informação, as redes de computadores, os bancos de dados, sistema de energia e comunicação são um dos pontos de vulnerabilidade e risco.

Os invasores necessitam de alguma motivação ou objetivos pessoais para desejarem adentrar numa rede sem a devida permissão, sejam elas pessoais ou financeiras.

5.3.1 Segurança em Redes de Computadores

Deve ser considerado o que diz Torres (2010), que a segurança preventiva implementa ações que procuram evitar que dados sejam danificados ou comprometidos sem que a ação direta de terceiros ou sem que a ação de pessoas mal intencionadas origine o problema. Segundo Santo (2010) para obter segurança em uma aplicação para Internet ou Intranet, é preciso cuidar de quatro elementos básicos:

- Segurança na estação (cliente);
- Segurança no meio de transporte;
- Segurança no servidor
- Segurança na Rede Interna.

De acordo com Stapko(2007), segurança de computadores consiste em proteger informações pessoais ou confidenciais e/ou recursos computacionais de indivíduos ou organizações que poderiam deliberadamente destruir ou utilizar tais informações para fins maliciosos. Algumas propriedades devem ser garantidas para uma completa e eficaz implementação de segurança em sistemas computacionais (Kurose *et al.*, 2009; Stallings, 2008; Bishop, 2005):

Confidencialidade: trata-se da ocultação de informações ou de recursos, protegendo-os contra acesso não autorizado. Um exemplo prático pode ser observado em instituições de ensino, onde o acesso à informação é restringido de acordo com classes de usuários. Existe ainda a preocupação em manter secretas as informações pessoais de cada aluno. Confidencialidade, portanto, é a garantia de que somente o remetente e o destinatário pretendido terão o poder de entender o conteúdo da mensagem. Se algum intruso conseguir interceptar a mensagem, não deverá conseguir extrair informações do texto cifrado (disfarçado, ou ilegível).

Autenticidade: é a garantia de que a entidade participante da comunicação é realmente quem ela afirma ser. Remetente e destinatário precisam confirmar a identidade mútua. Quando a comunicação se dá pessoalmente entre seres humanos, esse problema é facilmente solucionado por reconhecimento visual. O problema existe quando a comunicação não permite que as partes sejam vistas (caso dos sistemas computacionais).

Integridade: integridade se refere à confiabilidade dos dados ou recursos, ou seja, trata-se da garantia de que não houve mudanças durante a comunicação (ou seja, não contêm modificação, inserção, exclusão ou repetição). Extensões das técnicas de soma e verificação encontradas em protocolos de transporte e de enlace confiáveis podem ser utilizadas para proporcionar integridade à mensagem.

Não repúdio de mensagem: o receptor pode, ainda, comprovar que a mensagem veio de um remetente específico. Trata-se de uma proteção de negação, por parte de uma das entidades envolvidas na comunicação, de ter participado de parte ou de toda a comunicação, propriedade conhecida como não-repúdio (ou irretratabilidade).

Disponibilidade: disponibilidade refere-se à capacidade de acesso a informações e serviços sempre que necessário, ou seja, um sistema estará disponível se oferecer os serviços, de acordo com o projeto do sistema, sempre que os usuários os solicitarem.

Segundo Stallings (2008), ameaça pode ser definida como um potencial para violação da segurança quando há uma circunstância, capacidade, ação ou evento que pode quebrar a segurança e causar danos. Ou seja, uma ameaça é um possível perigo que pode explorar uma vulnerabilidade.

O intruso de um sistema pode, muitas vezes, não só escutar o que se passa no canal de comunicação intruso passivo, como também pode gravar mensagens e reproduzi-las mais tarde, injetar suas próprias mensagens ou modificar mensagens legítimas antes que elas cheguem ao receptor intruso ativo (Tanenbaum, 2003).

Segundo Kurose (2009) as propriedades da comunicação dentro de uma rede segura são:

- Confidencialidade;
- Autenticação do ponto final;
- Integridade da mensagem;
- Segurança operacional.

Esses aspectos pregam que a mensagem deve ser entendida apenas por remetente e destinatário, e estes por meio de autenticação devem confirmar a identidade dos envolvidos na comunicação, assim garantindo que no processo de envio da mensagem a mesma não seja alterada ou se perca no meio do caminho e a segurança operacional.

A estratégia básica de segurança com o que foi apresentado é traçada por Tanembaum (2010) onde na camada de rede, podem ser instalados firewalls para manter ou descartar pacotes. A segurança do *IP* também funciona nessa camada. Na camada de transporte, é possível criptografar conexões inteiras fim a fim, ou seja, processo a processo. Para obter segurança máxima, é necessário que ela seja fim a fim. Finalmente, questões como autenticação do usuário e não repúdio só podem ser tratadas na camada de aplicação. Onde tudo isso será apresentado logo adiante.

5.3.1.1 Controles de acesso

Conforme Controle de Acesso (2007), controle de acesso em segurança, especificamente em segurança física de ambientes, é a permissão do acesso a recursos, salas, prédios, entre outros, somente pessoas autorizadas. O controle físico de ambientes é feito por pessoas, meios tecnológicos, cartão de acesso, abertura de porta por meio de tranca eletrônica e/ou liberado por senha, ou

mecanismos de segurança como: catracas, fechaduras, chaves, entre outros.

Segundo Campos (2006), o controle às informações deve atender ao determinado nível conforme os requisitos de segurança, sempre contribuindo com o negócio da organização. O controle de acesso na segurança da informação é baseado basicamente em três processos:

Autenticação, autorização e contabilidade. Assim sendo, pode-se dizer que o controle de acesso é a habilidade de permitir ou negar um objeto, sendo esse uma entidade passiva, um arquivo, um sistema, entre outros, por um sujeito, uma entidade ativa, sendo esse um usuário ou processo. A autenticação identifica quem acessou o recurso, a autorização define o que o usuário pode fazer e a contabilidade informa o que esse usuário fez:

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

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

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

5.3.1.2 Autenticação e autorização

Autenticação: é uma referência ao procedimento que confirma a validade do usuário que realiza a requisição de um serviço. Este procedimento é baseado na apresentação de uma identidade junto com uma ou mais credenciais.

As senhas e os certificados digitais são exemplos de credenciais.

Segundo Peres (2010), as redes locais sem fio WLAN estão especificadas no padrão da *IEEE* como *IEEE* 802.11 [1]. O padrão IEEE 802.11b, utilizado como foco neste artigo, opera utilizando freqüências entre 2.4*GHz* a 2.5*GHz ISM* com DSSS. Possuem a taxa de transferência de 11*Mbps*.

Para identificar as fraquezas nas implementações atuais, é necessária uma análise nos princípios de segurança existentes no padrão das redes sem fio. Serão apresentadas as características de autenticação de dispositivos, ou seja, da garantia de que uma determinada informação veio de um equipamento autorizado, e de privacidade, que garante a confidencialidade de informações trocadas entre os dispositivos.

O padrão IEEE 802.11 define duas formas de autenticação: *open system e shared key*. Independentemente da forma escolhida, toda autenticação deve ser realizada entre pares de estações, nunca havendo comunicação *multicast*. Em sistemas BSS as estações devem se autenticar e realizar a troca de informações através do *Access Point* [1]. As formas de autenticação previstas definem:

• Autenticação *Open System* - é o sistema de autenticação padrão sendo que, neste sistema, qualquer estação será aceita na rede, bastando requisitar uma autorização. É o sistema de autenticação nulo.

• Autenticação *Shared key* - nesta autenticação, ambas as estações (requisitante e autenticadora) devem compartilhar uma chave secreta. A forma de obtenção desta chave não é especificada no padrão, ficando a cargo dos fabricantes a criação deste mecanismo. A troca de informações durante o funcionamento normal da rede é realizada através da utilização do protocolo *WEP*.

A autenticação do tipo *Open System* foi desenvolvida focando redes que não necessitam de segurança para autenticidade de dispositivos. Nenhuma informação sigilosa deve trafegar nestas redes já que não existe qualquer proteção. Também aconselha-se que estas redes permaneçam separadas da rede interna por um *firewall*.

A autenticação *Shared Key* utiliza mecanismos de criptografia para realizar a autenticação dos dispositivos. Um segredo é utilizado como semente para o algoritmo de criptografia do *WEP* na cifragem dos quadros. A forma de obter esta autenticação é a seguinte:

1. Estação que deseja autenticar-se na rede envia uma requisição de autenticação para o *AP*.

2. O *AP* responde a esta requisição com um texto desafio contendo 128 *bytes* de informações pseudo-randômicas.

3. A estação requisitante deve então provar que conhece o segredo compartilhado, utilizando-o para cifrar os 128 *bytes* enviados pelo *AP* e devolvendo estes dados ao AP.

4. O *AP* conhece o segredo, então compara o texto originalmente enviado com a resposta da estação.

Autorização é o processo de dar permissão a alguém para fazer ou ter alguma coisa. Em sistemas de computador multiusuário, um administrador do sistema define para o sistema que os usuários têm permissão de acesso ao sistema e quais privilégios de uso (como o acesso a diretórios de arquivos que, horas de acesso, quantidade de espaço de armazenamento alocado, e assim por diante). Supondo que alguém tenha ligado para um computador sistema operacional ou aplicativo, o sistema ou aplicativo pode querer identificar quais recursos o usuário pode ser dado durante esta sessão. Assim, a autorização é às vezes visto como a definição preliminar acima de permissões por um administrador do sistema e a verificação real dos valores de permissão que foram criadas quando um usuário está tendo acesso.

Logicamente, a autorização é precedida de autenticação.

5.3.1.3 Criptografia

De acordo a *MCT* (2002), a Internet nasceuna década de 60 e foi fruto da engenhosidade de cientistas como Michael Dcitouzos, que criou os chamados roteadores, computadores que controlam e direcionam o tráfego na internet.

Após a Guerra Fria os americanos montaram uma rede onde não havia hierarquia, assim qualquer um podia acessá-la e caso algum dano fosse causado à rede esta continuaria em funcionamento. A Internet está presente em todo o mundo, sendo utilizada para os mais variados serviços, é também, a responsável pelo surgimento de novas profissões.

Hoje a internet está integrada na vida da população de forma mais ampla

e veio a se tornar um instrumento indispensável ao crescimento intelectual e econômico, está presente em diversas localidades do planeta. Segundo Gallo (2002), a definição de rede é: “conectar um grupo de sistemas com o propósito expresso de compartilhar informação. Os sistemas que se conectam formam uma rede.” Ainda de acordo a Gallo (2002), para a rede funcionar é necessário que se siga vários preceitos tais como: Comunicação entre computadores e tecnologia de rede; protocolo e metodologia de comunicação; topologia e projeto; endereçamento; roteamento; confiabilidade; interoperabilidade; segurança e padrões.

Segundo Tanenbaum (2003), em resumo, a criptografia pode ser entendida como um conjunto de métodos e técnicas para criptografar (cifrar ou codificar) informações legíveis por meio de um algoritmo de criptografia parametrizado por uma chave, convertendo um texto original, denominado texto aberto (texto claro ou texto simples), em um texto ilegível, denominado texto cifrado (cifra ou texto código). Posteriormente, é possível para o receptor decriptar este texto cifrado, ou seja, efetuar o processo reverso e recuperar as informações originais.

Segundo Silva (2013), as redes wireless abriram uma brecha enorme na segurança de sistemas em rede. Isso porque os dados podem ser facilmente interceptados com algum conhecimento técnico, isso obrigou o desenvolvimento de técnicas de criptografia para tornar esse tipo de comunicação viável, não só para empresas que decidem conectar seus usuários por meio de redes sem fio, mas, também, para que os usuários domésticos possam realizar suas transações financeiras com mais segurança e privacidade. Os tipos de criptografia mais usados nas redes wireless são:

WEP: provê cifração de dados e privacidade nas informações trafegadas pelas redes *wireless*. (Rufino, 2005). Utiliza o conceito de chaves compartilhadas *ou Shared Key* e processa os dados utilizando chaves idênticas em ambos os dispositivos de conexão. Para cifrar informações uma chave de64 ou 128 *bits* é utilizada, sendo desses valores 24 *bits* de um Vetor de Inicialização, que a cada pacote é alterado aleatoriamente para melhor proteger a chaves.

Esta técnica usa uma chave secreta compartilhada e o algoritmo de criptografia *RC4*. O roteador wireless ou ponto de acesso, bem como todas as estações que se conectam a ele devem usar a mesma chave compartilhada. Para

cada pacote de dados enviado em qualquer direção, o transmissor combina o conteúdo do pacote com uma soma de verificação desse pacote. O padrão *WEP* pede então que o transmissor crie um vetor de inicialização específico para o pacote, que é combinado com a chave e usado para criptografar o pacote. O receptor gera seu próprio pacote correspondente e o usa para decodificar o pacote. Em teoria, essa abordagem é melhor do que a tática óbvia de usar apenas a chave secreta compartilhada, pois inclui um bit de dado específico para o pacote que dificulta sua violação. Entretanto, se uma chave compartilhada estiver comprometida, um invasor poderá bisbilhotar o tráfego de informações ou entrar na rede.

WPA e *WPA2*: estes certificados de segurança são baseados no padrão da *Wi-Fi Alliance* para redes locais sem fio e utilizados por muitas empresas e até em redes domésticas. Eles permitem autenticação mútua para verificação de usuários individuais e criptografia avançada. A *WPA* fornece criptografia para empresas, e a *WPA2* considerada a próxima geração de segurança *Wi-Fi* vem sendo usada por muitos órgãos governamentais em todo o mundo.

O *WPA2* com *AES* é a novidade, tanto para o uso corporativo quanto para o pessoal. Ao usuário residencial, ele garante um excelente padrão de segurança e, aos usuários corporativos, permite agregar um servidor de autenticação para controle dos usuários em conjunto com a criptografia.

Chaves Simétricas

A criptografia de chave simétrica (ou criptografia de chave privada) possui este nome porque os processos de criptografia e decriptografia são realizados com uma única chave, ou seja, tanto o emissor quanto o receptor detêm a mesma chave e esta deve ser mantida em segredo para que se possa garantir a confidencialidade das mensagens ou da comunicação.

Exemplos de algoritmos classificados como simétricos são o *DES* e o *AES* (*NIST*, 2001).

O texto legível é criptografado em texto cifrado pelo emissor utilizando uma chave secreta compartilhada. Após ser transmitida, a mensagem cifrada é então descriptografada pelo receptor utilizando a mesma chave secreta.

Figura 5 - Chave Simétrica
Fonte: Google Imagens = Chave Simétrica

Segundo Moreno (2005), a principal vantagem da criptografia de chave simétrica é que os algoritmos deste tipo são rápidos e podem operar em tamanhos arbitrários de mensagens. Em contrapartida, a desvantagem está na dificuldade de gerenciamento da chave compartilhada, a qual deve ser enviada de modo seguro a todos os usuários autorizados antes que as mensagens possam ser trocadas e ainda deve ser mantida em segredo.

Segundo Just (2010) os algoritmos simétricos podem ser classificados em duas categorias: de fluxo e de bloco. Os algoritmos de fluxo operam em fluxos de dados criptografando símbolo por símbolo, ou seja, um *bit* de cada vez. Por outro lado, os algoritmos de bloco operam sobre blocos de tamanho fixo e pré-definido. Nos algoritmos de fluxo, uma chave de criptografia é utilizada para um símbolo apenas, enquanto nos de bloco, uma única chave é utilizada para todos os símbolos de um bloco.

Chaves assimétricas

Para Stallings (2008), a criptografia assimétrica, mais conhecida como criptografia de chave pública, utiliza um par de chaves denominadas chave privada e chave pública. Qualquer uma das chaves pode ser utilizada para criptografar os dados, porém a mesma não pode ser utilizada para decriptografá-los, isto é, se a criptografia for realizada com a chave pública, somente a respectiva chave privada poderá realizar a decriptografia, ou vice-versa. Para que este tipo de criptografia obtenha sucesso é fundamental que a chave privada seja mantida em

segredo, enquanto a chave pública pode, e deve ser divulgada a outros usuários que desejam se comunicar.

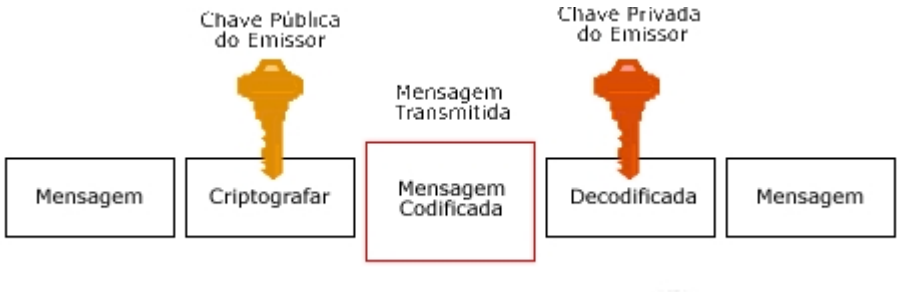

Figura 6 - Chave Assimétrica
Fonte: Google Imagens = Chave Assimétrica

Devido ao fato de uma chave ser pública e a outra ser mantida em segredo, um criptossistema de chave pública deve atender às seguintes condições (Bishop (2007)):

1. Deve ser possível criptografar ou decriptografar uma mensagem dada a chave apropriada;
2. Deve ser computacionalmente inviável derivar a chave privada a partir da chave pública;
3. Deve ser computacionalmente inviável derivar a chave privada por meio de ataques do tipo Texto Puro Escolhido.

O algoritmo *RSA* tornou-se praticamente um sinônimo de criptografia de chave pública, embora existam muitos outros algoritmos potencialmente utilizáveis (Kurose (2003)). O *RSA* é considerado um algoritmo forte, porém sua principal desvantagem é a lentidão com que opera, uma vez que se costuma aplicar chaves grandes para garantir uma segurança adequada. De acordo com Stapko (2007), uma das propriedades mais úteis do *RSA* é que ele pode ser usado tanto para a operação básica de chave pública (troca de mensagens entre duas entidades) quanto para autenticação (onde uma entidade criptografa uma mensagem com sua chave privada e a envia para alguém que detenha sua chave pública, garantindo que a origem é realmente do dono da chave privada). Uma aplicação interessante para o *RSA* está no protocolo *SSL*, que utiliza uma operação de chave pública e outra de autenticação.

Assinaturas Digitais

Segundo Silva (2013), um recurso conhecido por Assinatura Digital é muito usado com chaves públicas. Trata-se de um meio que permite provar que um determinado documento eletrônico é de procedência verdadeira.

Quem recebe um documento assinado digitalmente usa a chave pública fornecida pelo emissor para se certificar da origem. Além disso, a chave é integrada ao documento, isso implica que qualquer alteração realizada nas informações vai invalidar o documento. Ao procurar-se pelo entendimento tem-se que assinatura digital é firmar com seu nome digitalmente, sendo uma identificação composta por números. É um método que garante a chegada de uma mensagem sem ter sido alterada no seu caminho até seu destino final.

Exatamente como acontece com as assinaturas por escrito, a assinatura digital deve ser verificável, não falsificável e incontestável. Isto é, deve ser possível provar que um documento assinado por um indivíduo foi na verdade assinado por ele (verificável) e que somente aquele indivíduo poderia ter assinado o documento (a assinatura não pode ser falsificada e o signatário não pode mais tarde repudiar o documento, nem negar que o assinou) (Kurose (2003)). Normalmente, algoritmos de chave pública são empregados para alcançar soluções de assinatura digital.

Criptografia Quântica

Este tipo de codificação de informação difere dos demais métodos criptográficos porque não precisa do segredo nem do contato prévio entre as partes.

A criptografia quântica permite a detecção de intrusos e é incondicionalmente segura mesmo que o intruso tenha poder computacional ilimitado. Mas o seu custo de implantação é muito elevado. Outro fato limitante para a adoção dessa técnica é a taxa de erros na transmissão dos fótons, seja por ondas de rádio ou fibra ótica.

5.3.1.4 *Proxy*

Em sua grande maioria, os navegadores de páginas *web*, fazem conexões diretas com a Internet. Mas há outra forma bem mais interessante de conexão: eles podem ser configurados para se conectarem através de um servidor *Proxy*.

O *Proxy* é um serviço que está disponível em um ambiente servidor, que recebe requisições das estações de trabalho para conexões à Internet, onde seu papel fundamental é buscar a informação primeiramente no seu *cache* local e caso não encontre o documento requisitado, faz a busca no site solicitado pela estação de trabalho. Na segunda situação, o endereço Internet que fica registrado no servidor da página solicitada, é o do servidor *Proxy*, pois o mesmo é o dispositivo que está entre a rede local e a Internet (P*roxy*, 2007).

Conforme Equipe Conectiva (2001), o servidor *Proxy* surgiu da necessidade de ligar a rede local à grande rede de computadores, a Internet, através de um computador que provesse o compartilhamento de Internet com os demais computadores. Pode-se fazer a seguinte analogia: rede local é uma rede interna e a Internet é uma rede externa, sendo assim, o *Proxy* é o dispositivo que permite as máquinas da rede interna se conectarem ao mundo externo. Como na maioria dos casos as máquinas da rede local não têm um endereço válido para a Internet, elas fazem a solicitação de um endereço externo para o servidor *Proxy*, que encaminha a requisição à Internet. Caso não ache o documento solicitado em seu *cache* de Internet, o servidor está habilitado a fazer essa consulta, pois o mesmo tem um endereço válido na Internet. Sendo assim, pode-se dizer que é normal ter um servidor *Proxy* diretamente ligado à Internet e com um endereço válido.

Segundo Ricci e Mendonça (2006), o servidor *Proxy* pode ser definido como um software que atua como gateway de aplicação entre o cliente e o serviço que é acessado.

O servidor *Proxy* intercepta as requisições do cliente enviadas ao servidor, as interpreta e então repassa as requisições ao servidor de destino responsável pelo serviço a ser acessado, realizando o mesmo procedimento com a resposta.

Aprofundando-se mais no servidor *Proxy*, Ricci e Mendonça (2006) ressaltam que o servidor *Proxy* é capaz de analisar os pacotes de rede na camada de aplicação, ou camada 7, do modelo OSI ou Open Systems

Interconnection. Dessa forma é possível oferecer uma grande flexibilidade pois permite que o tráfego de dados de um serviço possa ser analisado e assim permitindo diversos tipos de ações, como por exemplo a aplicação de filtros de bloqueio, ou então o registro dos dados trafegados (para fins estatísticos e/ou de monitoramento, por exemplo).

Segundo Gualberto e Silva (2007) o *Proxy* tem a função de concentrar todas as requisições das mais diversas origens, canalizando-as por uma mesma saída. Ele é que, efetivamente, faz a requisição ao destino. Funciona como um intermediário entre o cliente e o servidor de destino. Esse intermediário efetua tais requisições segundo regras, ou filtros, implementados pela ferramenta de *Proxy*. Tais filtros têm a função de proibir ou liberar acessos a sites, endereços identificadores de máquinas e redes, strings e até limitar velocidade de acesso. Um servidor *Proxy* é utilizado para gerir os direitos de acesso, largura de banda ou distribuir conteúdo *cache*. Os quatro principais tipos de *Proxy* são *Web, caching, reverter* e transparente.

Um *Proxy* é um servidor que intercepta a conexão do seu computador para o servidor que você deseja se conectar. Um servidor *Proxy* é utilizado para gerir os direitos de acesso, largura de banda ou distribuir conteúdo *cache*. Os quatro principais tipos de *Proxy* são *Web*, *caching*, reverter e transparente. Alguns servidores têm proxies que incluem várias funções, tais como *Web* e *cache* ou reverter e *cache. Web Proxy.*

O servidor ao qual você deseja se conectar só vê o servidor *Proxy* como sendo ligado e não o seu computador. Os endereços *IP*, endereço único que se identifica na rede, permanecem desconhecidos para o servidor e os mesmos aumentar o anonimato do uso.

Cache

Conforme Watanabe (2000), o *cache* é onde os arquivos requisitados pelo servidor *Proxy* são armazenados e repassados posteriormente para os clientes, que são as estações de trabalho da rede interna. Esse é um aspecto que deve ser monitorado sempre, pois pode deixar um servidor inoperante, já que são arquivos armazenados em disco e caso falte espaço em disco o servidor não vai mais

funcionar. Para que isso não aconteça é necessário determinar quando os objetos serão atualizados ou removidos do *cache*, sendo que alguns desses podem permanecer sem alteração alguma por tempo indeterminado e outros podem sofrer alterações frequentemente.

Conforme *Proxy* (2007), visando o controle do *cache*, os servidores *Proxy* utilizam algoritmos de substituição que monitoram os objetos conforme seu cabeçalho, que contém a informação de período, tamanho e histórico de acessos. Dois deles são o *Least Recent Used,* que remove objetos existentes a muito tempo e o *Least Frequently Used*, que remove os objetos menos utilizados. A utilização do espaço em disco pelo *cache* do *Proxy* é controlada através desses algoritmos, juntamente com regras pré-determinadas pelo administrador.

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

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

• *Proxy Cache* – conforme *Proxy* (2007), são as implementações mais utilizadas de *Proxy*, e são conhecidos também como *caching web Proxy*. Este disponibiliza em *cache* páginas e arquivos de servidores remotos da Internet, permitindo que os clientes da rede local acessem de forma rápida esses arquivos, considerando que a velocidade do link da *LAN* é muito maior do que o com a Internet. Quando o *Proxy cache* recebe uma solicitação de acesso a um recurso externo, como uma página da Internet, este procura primeiramente em seu *cache* local e caso não encontre o recurso solicitado, ele imediatamente faz a requisição à Internet armazenando em seu *cache* e respondendo a solicitação do cliente. Por este motivo pode-se afirmar que a web *Proxy*, além de prover segurança, provê também alto desempenho para o acesso à Internet e permite criar filtros,

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

Uma das principais finalidades de muitos *proxies* é o *caching*. Onde o armazenamento de uma resposta obtida anteriormente para uso mais tarde, quando os clientes solicitarem o mesmo recurso. O *cache* retoma a resposta se acreditar que ela ainda está fresca (ou seja, o servidor de origem teria aprovado o retorno da resposta). A função de *caching* de um *Proxy* é opcional, ou seja, um *Proxy* realiza o papel de um *cache* além do seu papel como servidor para os clientes que estão por trás dele e como cliente para os servidores de origem.

Proxy transparente

Neste Modelo de configuração, os clientes não necessitam e nem devem configurar o uso do *Proxy*, pois as conexões *web* serão redirecionadas ao *Proxy* de forma transparente (automaticamente) (JUCÁ, 2005). É necessário utilizar o *iptables* para que basicamente as portas 80 e 443 sejam redirecionadas a porta do *Proxy* (geralmente, 3128). Dessa Forma, diferente do modelo convencional não é necessário configurar o *Proxy* manualmente em todos os computadores para navegar.

Conforme *Proxy* (2007) é uma forma de obrigarem os clientes a utilizarem o *Proxy*, ou seja, além das características do *Proxy cache*, ele implementa de forma transparente, por isso o nome, políticas de utilização e permite a coleta de dados estatísticos, entre outros. A transparência é implementada com a técnica de encaminhamento de portas, que é uma regra feita diretamente no *firewall* que faz o redirecionamento de todo o tráfego, por exemplo, *HTTP*, porta 80, para o *Proxy*. Sendo assim não importa as configurações do usuário, pois sua utilização estará sempre condicionada a política de acesso pré-determinada. O *Request* For *Comments* 5 3040, define esse método como *Proxy* interceptador.

Autenticação do *Proxy* pelo *LDAP*

Segundo Trigo (2007) o Protocolo Leve de Acesso a Diretório ou *LDAP* foi desenvolvido e padronizado em julho de 1993.

Com o passar dos anos, várias aplicações começaram a dar suporte de acesso a diretórios. O conceito de diretório é definido por Trigo (2007) como “algo para indicar direções”. A partir desse conceito, nota-se que o serviço de diretório serve como um indicativo para localizar a informação desejada. Arkills (2003), por sua vez, afirma que o serviço de diretórios, além de uma ferramenta eficiente na localização das informações, deve proporcionar o gerenciamento dessas informações tendo como princípio a centralização, pois, se existirem muitas fontes de informações, estas, em contrapartida, podem estar desatualizadas. No *LDAP*, a recuperação de informações é iniciada pela raiz da árvore e o dispositivo de busca vai percorrendo os nós até encontrar o elemento desejado. Segundo Trigo (2007), a raiz e os ramos da árvore são diretórios. Cada diretório pode conter outros diretórios ou elementos que são chamados de entradas; cada entrada possui um ou mais atributos que, por sua vez, podem ter um ou mais valores associados a eles, todos de acordo com um tipo de dados predefinido.

Segundo Trigo (2010) a inserção dos dados na base *LDAP* é feita através de arquivos LDIF ou *LDAP Data Interchange Format*, de texto puro, e que permitem a importação e alteração de dados, o backup e a replicação do diretório.

Segundo Evaristo (2008) as principais características de serviço de diretórios são:

a) auxiliar na organização das informações, centralizando em único repositório;

b) permitir que as informações sejam gerenciadas;

c) serviços de rede podem utilizar a informação centralizada em um repositório.

Servidor de *Proxy* w*eb*

Segundo Evaristo (2008) o serviço de *Proxy Web* é realizado através do serviço *Squid*, serviço padrão quando é necessário administrar o acesso à Internet. O *Squid* define um conjunto de regras, conhecidas também como *ACL* que

informam ao servidor *Squid* os acessos ou bloqueios que são permitidos.

Tem como função a autenticação *Proxy* e o histórico *cache* dos acessos realizados. Como *Proxy*, ele funciona como um intermediário com as transações realizadas na *web*, pois, aceita a requisição de um determinado cliente, faz o processamento e encaminha ao servidor. A requisição pode ser aceita, rejeitada ou até modificada antes do cliente receber a resposta do servidor. Como *cache*, armazena o conteúdo acessado recentemente para possível reutilização, pois, caso seja acessado o mesmo conteúdo, não há necessidade do *cache* recarregar a página desde o início.

O *Squid* apresenta algumas características, dentre elas destacam-se:

a) permite o gerenciamento das regras de acesso e bloqueios;

b) oferece estatísticas sobre o tráfego da *web*;

c) permite que somente os usuários autorizados possam navegar na internet;

d) Segundo Wessels (2004) reduz a carga de processamento do servidor *web* através do *cache*.

Autenticação do s*quid*

O funcionamento do serviço *Squid* em uma rede é ilustrado na figura abaixo. Nesta, o serviço aceita a requisição *HTTP* do cliente *LDAP* e consulta os servidores *HTTP* e *FTP* para conceder o acesso.

Squid

O *Squid* é um servidor *Proxy* de código-fonte aberto onde seu projeto iniciou-se em 1996 e utilizou-se como base o código-fonte do *software Harvest cache project* (Wessels, 2004).

O *Squid* é um Servidor *Proxy cache* de alto desempenho que suporta os protocolos *HTTP*,*FTP*, *TLS*, *SSL*. Reduz o uso da banda e melhora os tempos de resposta de páginas solicitadas que estão em *cache*. O *Squid* tem um grande controle de ACLS sendo muito flexível. Ele aumenta a velocidade de entrega da página solicitada ao cliente. Funciona em Linux, Unix e Windows, é licenciado sob o *GNU GPL.*

Wessels (2004) ressalta ainda que o *Squid* é um servidor *Proxy* bastante popular e oferece também a funcionalidade de *cache* de conteúdo. Wessels (2004) destaca alguns dos principais usos do *Squid*:

• Economizar banda do provedor de Internet enquanto se navega na *web*;

• Diminuir o tempo que uma página leva para carregar;

• Coletar estatísticas sobre o tráfego *web* da rede;

• Bloquear o acesso dos usuários à páginas inapropriadas conforme a política de uso da empresa;

• Garantir que apenas usuários autorizados possam navegar na Internet.

Proxy quer dizer intermediário, o *SQUID* funciona sendo o "atravessador" entre a conexão do cliente e o servidor, neste meio do caminho ele armazena os objetos que foram solicitados e permite que as próximas requisições para os mesmos objetos possam ser respondidas por ele mesmo.

O *Squid* possui as seguintes características:

• *Proxy* e *cache* para *HTTP*,*FTP* e outros protocolos baseados em URL

• *Proxy* para *SSL*

• *Cache* Hierárquico

• Suporte para *Proxy* transparente

• Políticas de controle de acesso extremamente flexíveis

• *SNMP*

• Logs Avançados

• *DNS Cache*

Segundo Wessels (2004), o *Squid* armazena as informações chaves de acesso dos usuários aos sites em um arquivo chamado “access.log”. Esse arquivo é baseado por linhas, ou seja, cada linha corresponde à uma requisição *HTTP* de um cliente.

Wessels (2004) explica que o formato padrão do arquivo de registro de acessos do *Squid* consiste em dez campos. A cada requisição feita por um cliente à um servidor, é armazenada no arquivo “access.log” uma linha contendo dez campos com as informações chave a respeito da conexão.

Vantagens e Desvantagens de um *Proxy*

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

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

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

• Redução de latência – possibilita a maior velocidade a resposta de requisições que são feitas ao objeto que está no *cache* do *Proxy* e não diretamente ao servidor remoto;

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

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

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

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

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

6 DESENVOLVIMENTO PRÁTICO

6.1 INTRODUÇÃO

Os passos executados a seguir foram efetuados na distribuição Ubuntu *Server* 14.04 que foi baixada diretamente do site da mantenedora.

Para obtenção de acesso ao servidor, que segue intitulado como TCC, a senha para acesso sem privilégios é *masterkey* e a senha para executar comandos como *root* é palavra chave *masterkey*.

Após o sistema ser instalado, o sistema foi atualizado por meio de comandos permitidos ao root sudo apt-get update. Por meio dos comandos apt-get foram instalados os serviços, por exemplo: *SQUID* e *DHCP*.

Para fim de estudo utilizamos uma distribuição na qual achamos possível ser introduzida na realidade da instituição, considerado vários pontos e decisões de projeto, em virtude que escolhemos formar as três redes e continuar trabalhando com duas interfaces de rede no servidor.

Sendo assim dividimos da seguinte forma os esquipamentos por nós utilizados:

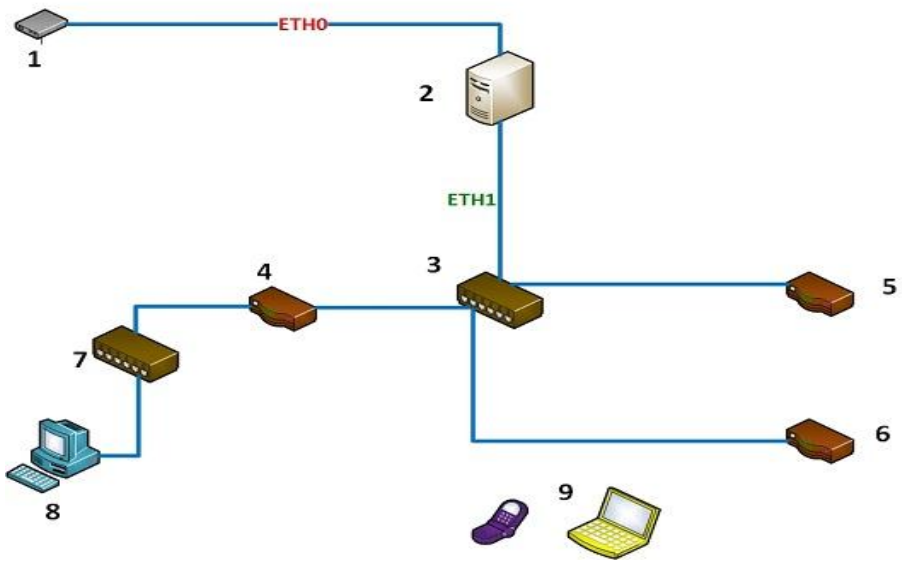

Figura 7 - Distribuição dos Equipamentos na Rede
Fonte: Autoria própria.

1 - Modem: recebe a internet do provedor e distribui para o servidor pela interface eth0;

2 - Servidor: no qual é instalado todos os serviços e que possui duas interfaces de rede, sendo elas eth0 e eth1;

3 - Switch 1: faz a distribuição da internet e dos serviços através do cabo de rede que está conectado na interface eth1 do servidor.

4 - Roteador 1: recebe a internet e os serviços e distribui. Possui algumas particularidades:

- Sem senha de acesso;

- *DHCP* desabilitado, recebe e distribui a faixa de *IP* que o servidor encaminha.

5 - Roteador 2: recebe a internet e os serviços e distribui. Possui algumas particularidades:

- Com senha de acesso, caso venha a ser configurada na instituição poderá ser requisitado na coordenação de Tecnologia da Informação;

- *DHCP* habilitado, distribui a faixa de *IP* que o roteador é configurado, pois como escolhemos trabalhar com duas interfaces de rede no servidor, não seria possível a configuração de distribuição de *IP* para as três redes distintas e com acessos distintos.

6 - Roteador 3: recebe a internet e os serviços e distribui. Possui algumas particularidades:

- Sem senha de acesso;

- *DHCP* habilitado, distribui a faixa de *IP* que o roteador é configurado, pois como escolhemos trabalhar com duas interfaces de rede no servidor, não seria possível a configuração de distribuição de *IP* para as três redes distintas e com acessos distintos.

7 - *Switch* 2: faz a distribuição da internet e dos serviços através do cabo de rede para os laboratórios da instituição;

8 - Computador *Desktop*: recebe a internet e os serviços provenientes do servidor configurado.

9 - Dispositivos Móveis: dispositivos que podem ser conectados em qualquer rede desde que tenha acesso.

Para ilustração dos comandos utilizados no terminal foi utilizado o carácter "#". Por exemplo: # sudo apt-get update.

Para ilustração dos comandos a serem inseridos dentro de arquivos de configuração a letra estará em Itálico e também dentro do arquivo com o carácter "#" são os comentários para uma melhor visualização do que está configurado.

6.2 EQUIPAMENTOS UTILIZADOS

SERVIDOR:(Do protótipo domiciliar)

- Microcomputador:
- Arquitetura x86;
- Processador Pentium 4 1.9 Ghz;
- Capacidade 1024 Mb de memória RAM;
- HD de 40 GB para armazenamento de dados;
- Placa de rede onboard;
- Placa de rede offboard;

REDE:

- Três Roteadores TPLINK Wireless N 300mbps;
- Um Switch TPLINK 12 portas;
- Um Switch TPLINK 48 portas;

6.3 INTERFACE DE REDE

6.3.1 Introdução

As interfaces de rede permitem roteamento e acesso remoto para que haja a comunicação de computadores através redes públicas ou privadas. As interfaces de rede têm dois aspectos relacionados: o hardware, o adaptador de rede, por exemplo, e a configuração de interface de rede que é o que será abordado.

O servidor possui duas unidades de rede, a denominada eth0 está conectada externamente, recebendo o link dedicado de internet. A eth1 está a nível local para a conexão interna e por isso está ligado ao roteador para que o sinal seja distribuído localmente.

6.3.2 Configuração

Os comandos para execução são:

 # sudo cd /etc/network

Dentro do diretório */etc/network* você encontrará um arquivo de configuração das interfaces, é o "interfaces"

 # sudo nano interfaces

Ao abrir o arquivo de configuração poderão ser feitas as alterações necessárias, então faça:

autolo
iface lo inet loopback

auto eth0
iface eth0 inet DHCP

auto eth1
iface eth1 inet static
address 192.168.10.5
netmask
255.255.255.0

auto eth1:2
iface eth1:3 inet static

address 192.168.20.5
netmask
255.255.255.0

auto eth1:3
iface eth1:3 inet static
address 192.168.30.5
netmask
255.255.255.0

Para salvar as alterações:

• *Ctrl + o* = salvar
• Opção sim = salvar alterações
• *Ctrl + x* = para sair

O arquivo final deve ficar como na imagem abaixo:

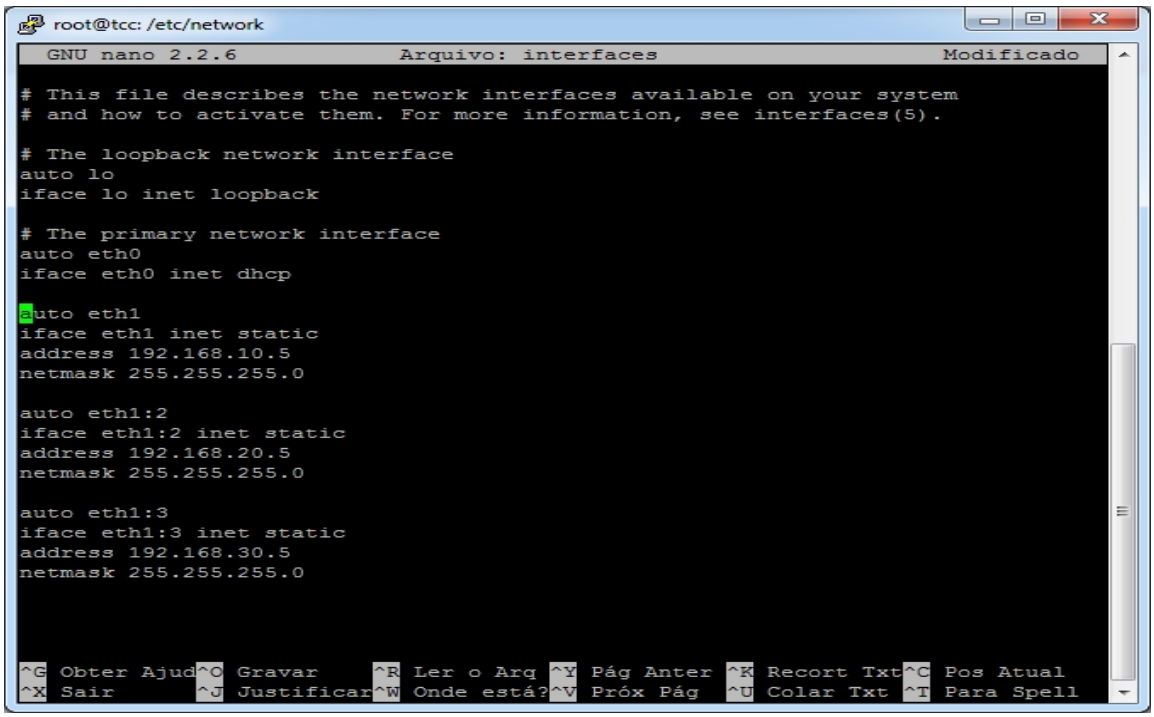

Figura 8 - Configuração Interfaces de Rede
Fonte: Autoria própria.

O serviço de rede deve ser reinicializado com o comando:

 # sudo /etc/init.d/networking restart

As placas de rede foram configuradas, sendo o primeiro bloco a configuração da interface de rede e *IP'S* dinâmicos para o que condiz o acesso a rede externa e a configuração da interface interna juntamente com um *IP* estático.

6.4 *FIREWALL*

6.4.1 Introdução

É o mecanismo criado para regular o tráfego entre redes diferentes e impedir a propagação de dados nocivos ou dados não permitidos em uma rede de computadores aplicando uma política de segurança.

Podem ser dispositivos físicos (hardware) ou softwares, ou ainda a combinação de ambos que é denominada Appliance. Os componentes do *firewall* são Chokes, Gates e Perimeter Network que também é conhecida como DMZ (De-Militarized Zone), Os principais tipos de *firewall* são *Proxy*, Packet Filter, Circuit Level, Gateways e Bastion Host.

Todo fluxo de rede, ou seja, todo o tráfego deve passar pelo *firewall*, caso contrário, existindo rotas alternativas, pode comprometer a segurança da rede.

Ter um Filtro de Pacotes no perímetro da rede, localizado entre o roteador as estações de trabalho, ou mesmo na borda da rede interna com a externa, aumentando assim a proteção contra acessos indevidos e bloqueio global de tráfego indesejado.

Utilizar Firewalls no contexto da rede interna, assim isolando redes separadas, e distintas, para que não exista colisão ou mesmo interceptação do tráfego entre elas.

Para a criação de nosso *firewall* utilizamos um módulo residente no kernel das distribuições Linux chamado *Iptables*. O *Iptables* trabalha em conjunto com o *framework* de pacotes *Netfilter* que também está presente no Kernel do Linux.

IPTABLES

Na configuração do *Firewall* com o *iptables*, é preciso saber quais são as regras a serem utilizadas para rodar o *Firewall*:

Regras do *Firewall*

- *INPUT*: É utilizada quando o destino final é a própria máquina *firewall*.
- *OUTPUT*: Qualquer pacote gerado pela máquina *firewall* e que deva sair

para a rede será tratado pela regra OUTPUT.

-*FORWARD*: Qualquer pacote que atravessa o *firewall*, de uma máquina e direcionado à outra, será tratado pela *chain FORWARD*.

Basicamente o *IPTABLES* tem as seguintes políticas:

- *DROP*: Nega um pacote e não manda um pacote de volta para o emitente.

- *ACCEPT*: Aceita o pacote

- *REJECT*: Nega um pacote e manda um pacote de volta do *tipo host-unreachable* (*Host* Inalcançável)

Comandos Principais do *Iptables*

a) -A - Este comando acrescenta uma regra às existentes no sistema, ou seja, permite atualizar regras já existentes na estrutura do *firewall*.

b) -I - Este comando insere uma nova regra dentro das existentes no *firewall*. c) -D - Este comando exclui uma regra específica no *firewall*.

d) -P - Este comando define a regra padrão do *firewall*.

e) -L - Este comando lista as regras existentes no *firewall*.

f) -F - Este comando ZERA todas as regras criadas no *Firewall* (o chamado flush).

g) -h - Este comando mostrará o help, ajuda de comando.

h) -R- Este comando substitui um regra no *firewall*.

i) -C - Este comando basicamente checa as regras. j) -Z - Este comando zera uma regra específica.

k) -N - Este comando cria uma nova regra com um nome.

l) -X - Este comando exclui uma regra específica por seu nome.

Os parâmetros padrão do *iptables* são os seguintes:

a) -p! (protocolo) - define qual o protocolo *TCP/IP* deverá ser tratado. São eles: *TCP, DP* e *ICMP*

b) -s! (origem) / -d! (destino) - Define qual o endereço de origem (-S) e de destino (-) que a regra atuará. Este comando possui dois argumentos: endereço/máscara e porta. Ex.: -S 10.0.0.1/24 80.

c) -i! (interface) - define o nome da interface de rede onde trafegará os pacotes de entrada e saída do *firewall*. Muito utilizado em mascaramento e técnicas

de NAT. Exemplo: - W eth1.

d) -j! (ir para) - Serve para redirecionar uma ação desde que as regras sejam similares. e) -f!(fragmento) - Trata datagrama fragmentados.

Os comandos e os parâmetros são exatamente iguais aos do ipchains, sem tirar nem pôr.

Extensões

-sport[!] [port:port] -dport[!] [port:port] - Normalmente estas extensões são utilizadas com o comando -m do *iptables*. Trata-se de um direcionamento de porta(s) origem (-sport), para porta(s) destino (-dport). Pode-se inclusive definir um número padrão de portas para o acesso (port:port). Este comando pode ser utilizado tanto para portas TCP ou UDP.

-*mac-source*[!] endereço- especifica qual a placa de rede, através de seu endereço *MAC*, que irá transmitir pacotes através do *firewall*, limitado pela política do mesmo.

-*icmp-type*[1] tipo- Especifica quais os tipos de pacotes *ICMP* pode passar ou não pelo *firewall*, São eles:

Mensagem	Tipo	Código
Echo-request	8	0
Echo-reply	3	0
Source-quench	4	0
Time-exceed	11	0
Destination-unreachable	3	0
Network-unreachable	3	0
Host-unreachable	3	1
Protocol-unreachable	3	2
Port-unreachable	3	3

Com isto podemos bloquear alguns ataques do tipo *ping flood*, bloquear ping e etc:[!] -- *syn*- especifica o uso dos *bits ACK* e *FIN* em requisições *SYN TCP*.

Especificamente, a opção `-*m state*' aceita uma opção adicional `*--state'*, que é uma lista de estados de ativação separados por vírgula. (a flag '!' não indica a ativação desses estados). Esses estados são:

- *NEW* Um pacote que cria uma nova conexão.

- *ESTABLISHED* Um pacote que pertence a uma conexão existente (isto é, um pacote de resposta).

- *RELATED* Um pacote que está relacionado com (mas não faz parte de) uma conexão existente, como um *ICMP error*, ou (com o módulo *FTP* inserido), um pacote que é estabelecido por uma conexão de dados *FTP*.

- *INVALID* Um pacote que não poderia ser identificado por alguma razão: isto inclui execução fora da memória e erros de *ICMP* que não correspondam a nenhuma conexão existente. Geralmente estes pacotes devem ser barrados (*drop*).

6.4.2 Configuração

Em nossos testes foi utilizado o *firewall* para habilitar a transferência do que foi requisitado para as portas 80 e 443 para a porta 3128 do *Squid* e outras configurações básicas.

Para alterar acessamos o arquivo # sudo nano /etc/init.d/rc.local

modprobe
iptable_nat
modprobe
IP_nat_ftp
echo 1 > /proc/sys/net/ipv4/IP_forward
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
iptables -t nat -A PREROUTING -i eth1 -p tcp --dport 80 -j REDIRECT --to-port 3128 iptables -t nat -A PREROUTING -i eth1 -p tcp --dport 443 -j REDIRECT --to-port 3128

Para salvar as alterações:

•Ctrl + o = salvar

•Opção sim = salvar alterações

•Ctrl + x = para sair

O arquivo final deve ficar como na imagem abaixo:

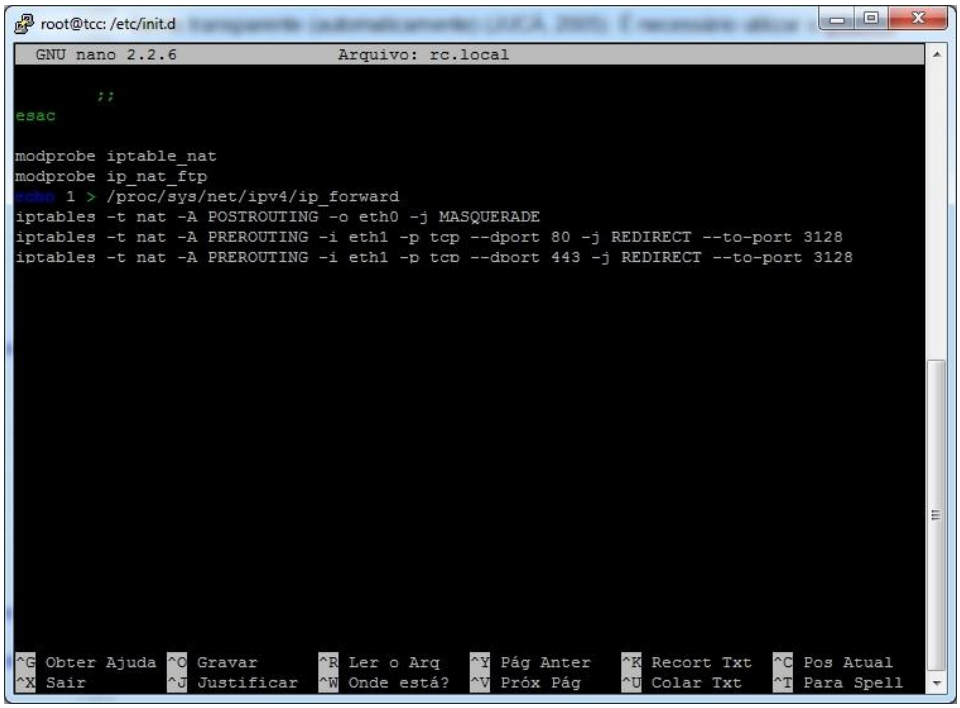

Figura 9 - Configuração *Firewall*
Fonte: Autoria própria.

Depois de alterado o documento, o servidor deve ser reiniciado.

6.5 *DHCP*

6.5.1 Introdução

DHCP é uma sigla usada no meio informático que significa Protocolo de Configuração Dinâmica de Endereços de Rede. Através do *DHCP* é possível fazer uma configuração automática e dinâmica de computadores que estejam ligados a uma rede *TCP/IP*.

O *DHCP* utiliza um modelo cliente-servidor, sendo que o servidor *DHCP* faz gestão centralizada (servido central) dos endereços *IP* que são usados na rede. O cliente *DHCP* consiste em um dispositivo de rede que tenha a capacidade de adquirir as configurações do TCP/*IP* de um servidor *DHCP*. Esse cliente tenta encontrar um ou mais servidores *DHCP* que ofereçam os padrões desejados para que o seu computador possa ser configurado de forma automática. O pacote enviado pelo servidor *DHCP* contém especificações do endereço *IP*,

máscara, gateway e servidores *DNS*.

Graças ao *DHCP*, os dispositivos da rede recebem as configurações do servidor central, sendo que dessa forma, o utilizador não precisa configurar os endereços manualmente. O *DHCP* consiste em um protocolo que é recomendado, porque auxilia e torna viável a gestão de grandes redes de *IP'S*, e facilita a vida de utilizadores que se desloquem muito com seus computadores portáteis.

Em nossas configurações utilizamos esse serviço para distribuir *IP'S* para um roteador, nos outros dois foi utilizado o serviço *DHCP* do próprio roteador em virtude de uma decisão de projeto de utilizarmos duas interfaces de rede em nosso servidor, eth0 e eth1, como já mencionado anteriormente.

6.5.2 Configuração

O comando utilizado para baixar e instalar o serviço é:

 # apt-get install isc-DHCP-server

Acessar o arquivo de configuração dhcpd.conf.

sudo nano /etc/*DHCP*/dhcpd.conf

Configurar do seguinte modo: INTERFACES="eth0"

default-lease-time

600; max-lease-time

7200; authoritative;

log-facility local7;

subnet 192.168.10.0 netmask 255.255.255.0{

RANGE 192.168.10.10 192.168.10.253;

option domain-name-servers 200.134.80.11, 200.134.80.13;

option routers 192.168.10.5;

}

Para salvar as alterações:

• Ctrl + o = salvar

• Opção sim = salvar alterações

• Ctrl + x = para sair

O arquivo final deverá ficar como a imagem a seguir:

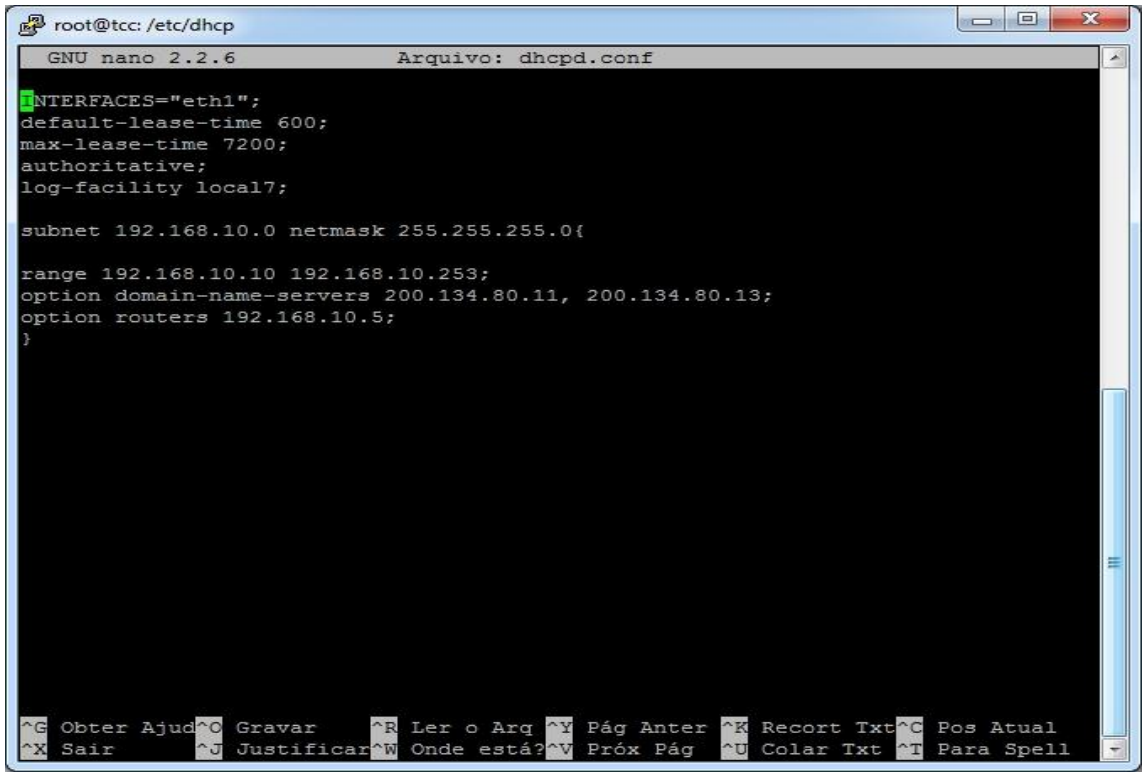

Figura 10 - Configuração* DHCP *Fonte: Autoria própria.

Como o Servidor *DHCP* será na interface eth1, deve-se editar o arquivo para configurar a interface eth1:

 # sudo nano /etc/default/isc-DHCP-server

Aonde está inserido eth0 mudar para eht1

INTERFACES="eth1"

Para salvar as alterações:

• Ctrl + o = salvar

• Opção sim = salvar alterações

• Ctrl + x = para sair

O arquivo final deverá ficar como a imagem a seguir:

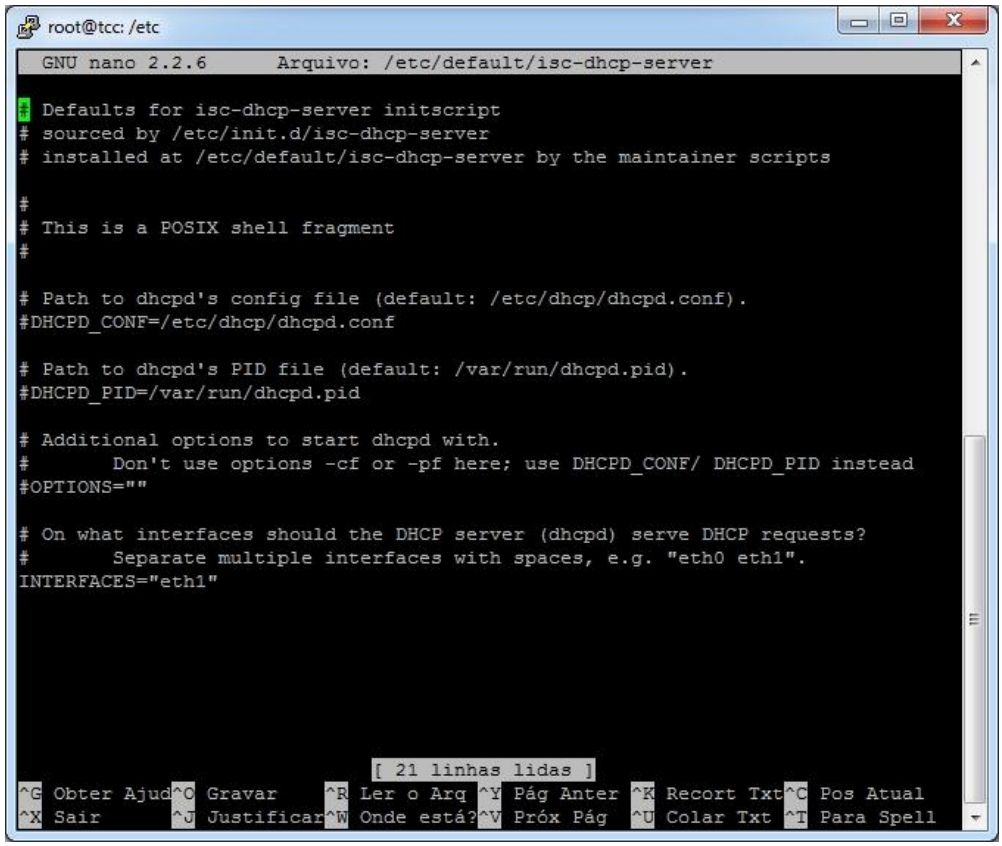

Figura 11 - Configuração Interface eth1

Fonte: Autoria própria.

Restart o serviço para concluir

 # sudo service isc-DHCP-server restart

6.6 *SQUID*

6.6.1 Introdução

Como já relatado anteriormente, o *Squid* trabalha com *ACLs* (Listas de Controle de Acesso) e através dessas listas de controle ele se torna uma poderosa ferramenta na administração de tráfego de conteúdo entre a rede interna e a externa.

6.6.2 Configuração

Instalar o *squid*.

 # sudo apt-get install squid3

O *Squid* será então instalado no diretório /etc/squid3. Após isto, é necessário editar o arquivo *squid.conf*, localizado dentro do diretório de instalação.

 # sudo nano /etc/squid3/squid.conf

```
dns_nameserver 200.199.252.72 200.200.252.132 8.8.8.8

acl SSL_ports port 443
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
http_port 192.168.10.5:3128 intercept
http_port 192.168.20.5:3128 intercept
http_port 192.168.30.5:3128 intercept
cache_dir ufs /var/spool/squid3 100 16
256 cache_log /var/log/squid3/cache.log
cache_mem 64 MB
visible_hostname TCC
acl bloquear url_regex -i "/etc/squid3/bloqueiasite"
acl bloquear2 url_regex -i
"/etc/squid3/bloqueiasite2" acl bloquear3 url_regex
-i "/etc/squid3/bloqueiasite3"
acl lab1 src
```

192.168.10.0/24 acl lab2

src 192.168.20.0/24 acl

lab3 src 192.168.30.0/24

http_access deny lab1 bloquear

http_access deny lab2

bloquear2 http_access deny

lab3 bloquear3

Para salvar as alterações:

- Ctrl + o = salvar
- Opção sim = salvar alterações
- Ctrl + x = para sair

O arquivo final deverá ficar como as imagens a seguir:

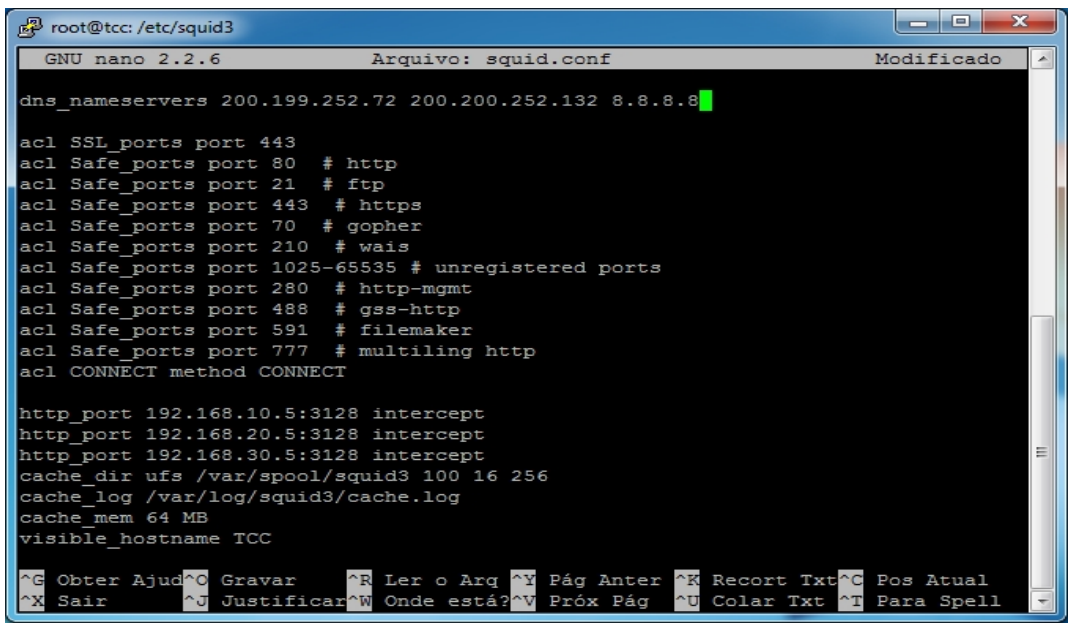

Figura 12 - Configuração *Squid* parte 1

Fonte: Autoria própria.

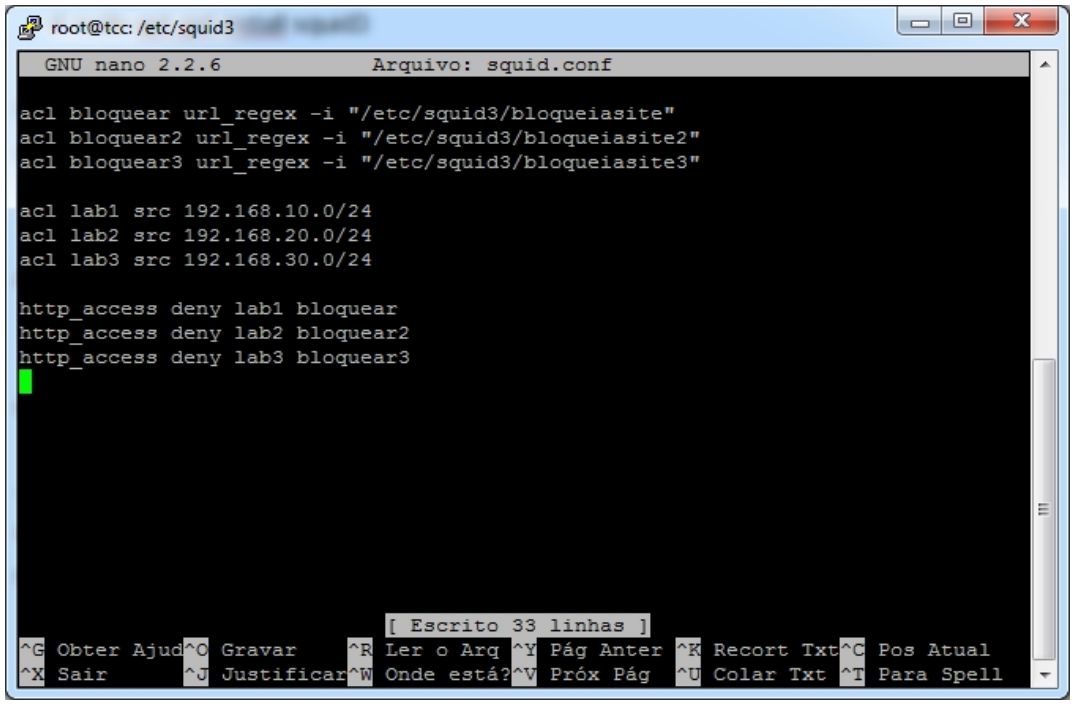

Figura 13 - Configuração *Squid* parte 2

Fonte: Autoria própria.

Arquivos necessários:

a) Arquivo de bloqueio no servidor

- Criar o arquivo com o nome e no diretório que colocamos dentro da ACL, que no nosso caso foi /etc/squid3/bloqueiasite. Para isso basta dar o comando:

#nano /etc/squid3/bloqueiasite

- A estrutura deste arquivo é muito importante para que ele seja um arquivo válido para o *Squid*. Ele só pode conter um site ou nome por linha, jamais coloque mais de um site por linha ou nome. Exemplo de arquivo com sites:

baixaki www.baixaki.com.br

Para salvar as alterações:

• Ctrl + o = salvar

• Opção sim = salvar alterações

• Ctrl + x = para sair

O arquivo final deverá ficar como a imagem a seguir:

Figura 14 - Arquivo de Bloqueio de Acesso
Fonte: Autoria própria.

Pronto, configuramos os arquivos que eram necessários serem configurados agora vamos levantar o serviço:

 # sudo /etc/init.d/squid start

7 COMPATIBILIDADE DE SISTEMAS

Para realização de testes de compatibilidade, foram utilizados computadores e dispositivos móveis com sistemas operacionais distintos, entre eles:

- Windows 7;
- Windows 8;
- Android 4.1.2;

Com os sistemas citados acima, foi detectada a rede sem fio, foi estabelecido uma conexão, com ips gerados de forma dinâmica pelo servidor, liberando o acesso a endereços *web* permitidos e negando para os pré-definidos anteriormente.

Windows 7:

Figura 15 - Compatibilidade Windows 7

Fonte: Autoria própria.

Windows 8:

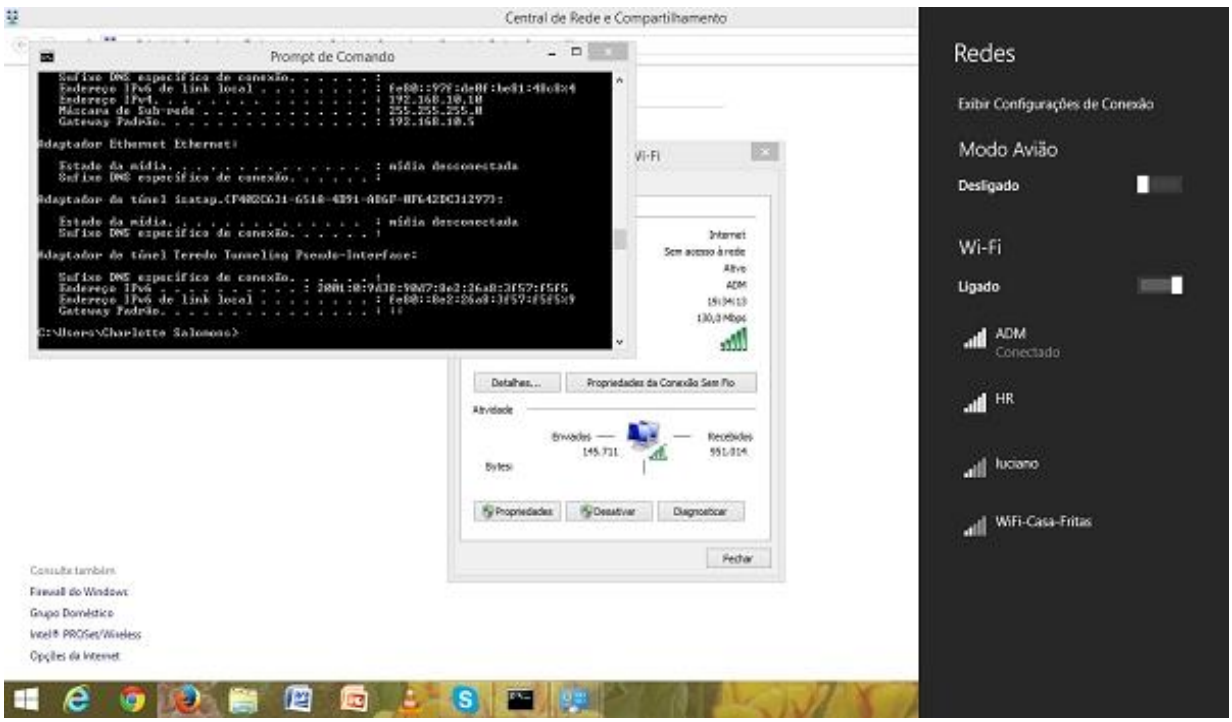

Figura 16 - Compatibilidade Windows 8

Fonte: Autoria própria.

Android:

Figura 17 - Compatibilidade *Android*
Fonte: Autoria própria.

8 COMPATIBILIDADE DE NAVEGADORES

Foram efetuados testes nos navegadores de internet para concluir o funcionamento mesmo sem a configuração manual do endereço do servidor *Proxy* ou por autenticação de usuário. Foram utilizados os três navegadores mais populares utilizados atualmente, com testes feitos na rede "ALUNO" e na rede "ADM", são eles Internet Explorer, Mozilla Firefox e Google Chrome. Os endereços solicitados tiveram o retorno esperado.

INTERNET EXPLORER:

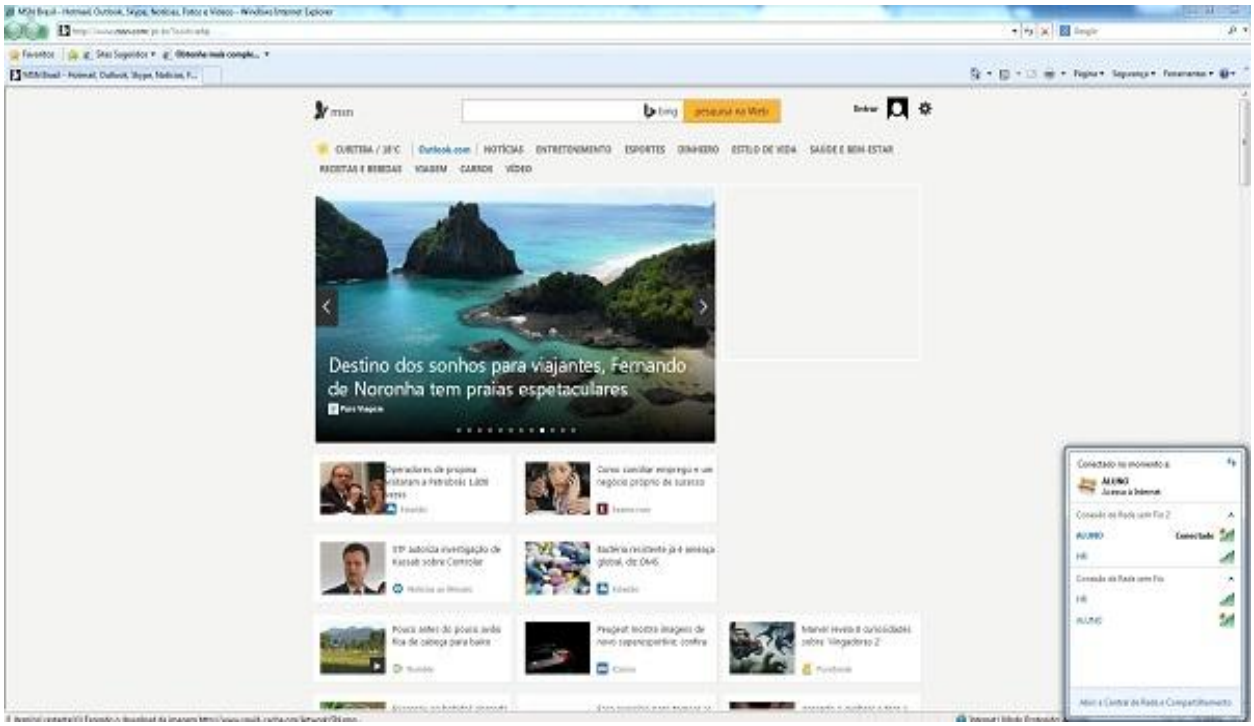

Figura 18 - Compatibilidade Internet Explorer
Fonte: Autoria própria.

MOZILLA FIREFOX:

- Compatibilidade Mozilla Firefox
Fonte: Autoria própria.

GOOGLE CHROME:

Figura 20 - Compatibiliade Google Chrome
Fonte: Autoria própria.

9 COMPARAÇÕES ENTRE REDES UTFPR E REDES SIMULADAS

A tabela a seguir mostra como se encontra as redes hoje na Universidade Tecnológica Federal do Paraná Campus Ponta Grossa e as redes as quais simulamos para realizarmos esse projeto.

Tabela 1 - UTFPR X REDE SIMULADA

	UTFPR	REDE SIMULADA
ALUNOS (SSID: ALUNOS)	Com acesso a rede UTFPRWEB, o usuário necessita inserir o registro acadêmico e senha previamente cadastrada para obter acesso a internet. Caso utilize rede cabeada, necessita indicar o endereço do servidor *Proxy*.	O usuário se conectará a uma rede aberta, sem que haja necessidade de autenticação, ou se estiver usando rede cabeada precise indicar o endereço do servidor *Proxy*.
ADMINIST./PEDAG. (SSID: ADM)	O usuário necessita inserir o endereço de um servidor *Proxy* e fazer alterações caso mude de rede, para que possa se conectar à internet.	Será necessário ao usuário saber apenas a chave de acesso à rede, que devido à restrição de acesso a funcionários, permite menos bloqueios aos usuários.
VISITANTES (SSID: VISITANTE)	Rede inexistente na configuração atual da instituição.	Rede com acesso controlado, mas que não é necessária senha para se conectar.

Fonte: Autoria própria.

10 TESTE DE ACESSO

Afim de testarmos as configurações efetuadas em todo o nosso projeto fizemos os testes de bloqueios de acesso já configurados anteriormente.

Navegador:

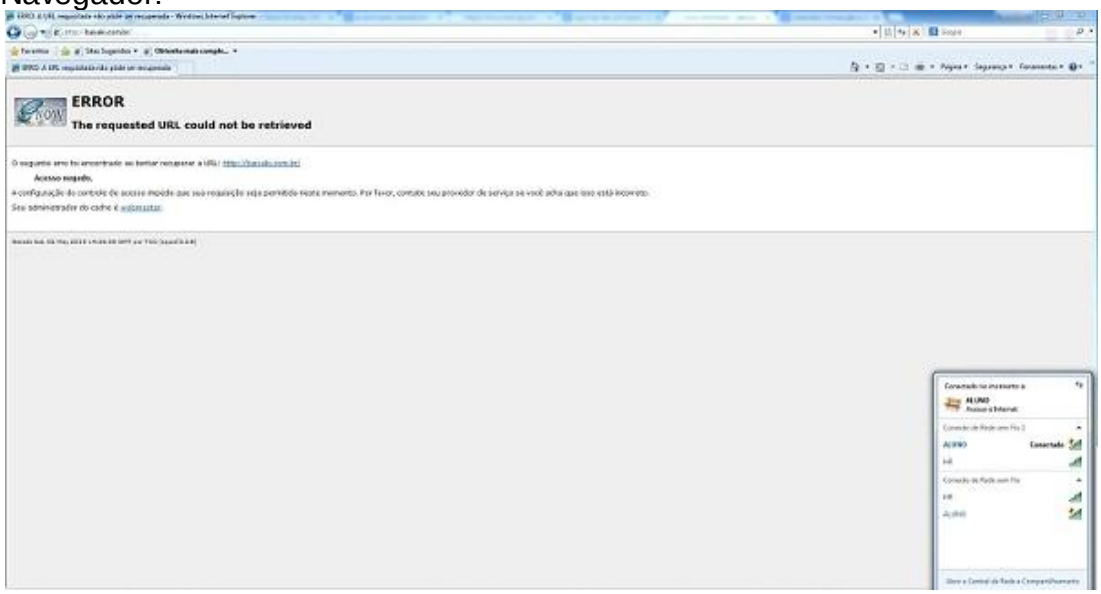

Figura 21 - Teste de Acesso Navegadores
Fonte: Autoria própria.

Dispositivos Móveis:

Figura 22 - Teste de Acesso Dispositivos Móveis
Fonte: Autoria própria.

11 CONCLUSÃO

Com base nos bem sucedidos testes executados nas redes ADM,ALUNO e VISITANTE, nos âmbitos sistemas operacionais e navegadores *web*, em todas as suas diversidades de versões, fica evidente que haveria a possibilidade de que o escopo de rede vigente poderia ser substituído pelo proposto no projeto de *Proxy* transparente. Levando-se em conta que hoje as redes vigentes de todos os campus do Paraná encaminham suas requisições até um servidor central no campus Curitiba, conclui-se que o projeto de *Proxy* transparente teria que ser adaptado para utilizar o mesmo modelo de centralização do fluxo de dados, porém o que foi projetado visa abolir a interdependência do que hoje está implantado. O servidor e ambientes de rede simulado, mesmo que em infinitamente em menor escala, inclusive no número de conexões simultâneas por parte dos usuários teste, ajudou na visualização de como seria o ambiente de rede da instituição, dando a percepção dos requisitos necessários para a criação das redes distintas usando um servidor *Proxy* transparente.

A proposição de não haver nenhum tipo de configuração ou identificação por parte dos usuários mostrou-se eficaz, o servidor *Proxy* fez a intermediação das requisições feitas pelo navegador *web*, para ambas as faixas de *IP'S* que diferenciavam as redes ADM, ALUNO e VISITANTE. A rede aluno não necessita de nenhum tipo de chave, ficando aberta para que possa ser utilizada como apoio didático ao aluno em seu período letivo. A rede ADM, focada na utilização dos setores administrativos e pedagógicos, possui chave para o acesso à rede, por se tratar de uma rede de cunho corporativo e necessitar de menos restrições. Todas as requisições *web* feitas pelos usuários, desviam de suas portas de comunicação e vão para a porta pré-configurada para que passem pelo *Proxy*, que dentro de seus scripts de configurações no *Squid*, possui uma série de diretrizes de segurança que mantém a segurança da rede.

REFERÊNCIAS

ARKILLS, B. ***LDAP* Directories Explained: An Introduction and Analysis**. Estados Unidos: Addison-Wesley, 2003. 432p.

BISHOP, M.; WAGNER, D. **Risks of e-voting: Comunications of the ACM**. Estados Unidos: ACM, 2007. 120p.

CAMPOS, A. L. N. **Sistema de segurança da informação: controlando os riscos**. Florianópolis: Visual Books, 2006. 180p.

CANTÚ, E. **Redes de Computadores e Internet**. São José: CEFET-SC São José, 2003. 79p.

EVARISTO, L. R. A. **Integrando a Base de Usuários *LDAP* com Serviços de E-MAIL, *Proxy Web* e SSH**. 2008. 53f. Monografia (Bacharelado em Redes de Computadores) - Sociedade Educacional de Santa Catarina, Santa Catarina, 2008.

GUALBERTO, R. P.; SILVA, R. B. **Controle Eficaz do Acesso à Internet**. 2007. 54f. Monografia (Especialização em Redes de Computadores) - Faculdade Selesiana de Vitória, Espírito Santo, 2007.

JUCÁ, H. L. **Técnicas Avançadas de Conectividade e *Firewall*: em GNU/Linux**. Rio de Janeiro: Brasport, 2005. 432p.

KUROSE, J. F.; MARQUES, A. S.; ROSS, K. W. **Redes de Computadores e a Internet: uma nova abordagem**. Estados Unidos: Addison Wesley, 2003. 548p.

KUROSE, J. F.; ROSS, K. W. **Redes de Computadores Uma Abordagem Top Down**. 5. ed. São Paulo: Pearson, 2009. 615p.

LOPES, R. V.; NICOLLETTI, P. S.; SAUVÉ, J. P. **Melhores Práticas para a Gerência de Redes de Computadores**. 1. ed. Rio de Janeiro: Campus, 2003. 408p.

MARCELO, A. ***Squid*: configurando o *Proxy* para Linux**. 4. ed. Rio de Janeiro: Brasport, 2005. 73p.

MENDES, D. R. **Redes de Computadores - Teoria e Prática**. São Paulo: Novatec, 2007. 384p.

MENDONCA, N.; RICCI, B. ***Squid* Solução Definitiva**. Rio de Janeiro: Ciência Moderna, 2006. 152p.

MONTEIRO, E. S. **Segurança no Ambiente Corporativo**. 1. ed. Florianópolis: Visual Books, 2003. 196p.

PERES, A. **Mecanismo de Autenticação Baseado na Localização de Estações Sem Fios Padrão IEEE 802.11**. 2010. 87f. Monografia Especialização em Redes de Computadores) - Universidade Federal do Rio Grande do Sul, Rio Grande do Sul, 2010.

RUFINO, N. M. O. **Segurança em Redes sem Fio**. São Paulo: Novatec, 2005. 208p.

SANTO, A. F. S. E. **Segurança da Informação**. 2010. 11f. Monografia (Bacharelado em **Segurança da Informação**) - Instituto Cuiabano de Educação, Mato Grosso, 2010.

SAUVÉ, J. P. **Gerência de Redes de Computadores**. 2002. 90f. Dissertação (Mestrado em Redes de Computadores) - Universidade Federal de Campina Grande, Paraíba, 2002.

SILVA, A. P. S.; SOUZA, T. M. X. **Criptografia em Redes de Computadores**. 2013. 5f. Monografia (Bacharelado em Rede de Computadores) - Universidade Paranaense, Paraná, 2013.

STALLINGS, W. **Criptografia e segurança de redes: princípios e práticas**. São Paulo: Pearson, 2008. 492p.

STAPKO, T. **Practical Embedded Security: Building Secure Resource-Constrained Systems (Embedded Technology)**. 1. ed. Estados Unidos: Newnes, 2007. 284p.

TANENBAUM, A. S. **Redes de Computadores**. 4. ed. Rio de Janeiro: Campus, 1997. 968p.

TANENBAUM, A. S.; WETHERALL, D. **Redes de Computadores**. 5. ed. São Paulo: Pearson, 2011. 600p.

TORRES, G. **Redes de Computadores Curso Completo**. 1. ed. Rio de Janeiro: Axcel Books, 2001. 653p.

TRIGO, C. H. **OpenLDAP - Uma Abordagem Integrada**. São Paulo: Novatec, 2007. 240p.

WESSELS, D. ***Squid*: The Definitive Guide**. Sebastopol, CA, Estados Unidos: O'Reilly Associates, 2004. 442p.

Comunicação entre Computadores e Tecnologias de Rede.. M. A Gallo, 2002.Disponível em: <www.books.google.com.br> Acesso em: 8 dez. 2014

Considerações sobre Segurança em Redes sem Fio. Disponível em: <http://www.redes.unb.br/ceseg/anais/2003/07.pdf> Acesso em: 22 abr. 2015

Controle de Acesso. Disponível em: <http://pt.wikipedia.org/wiki/Controle_de_acesso> Acesso em: 22 out. 2014

Ferramentas de IDS. Disponível em: <https://memoria.rnp.br/newsgen/9909/ids.html> Acesso em: 1 mai. 2015

Google Imagens. Disponível em: <https://www.google.com.br/search?q=chaves+simetricas&espv=2&biw=1366&bih=643&source=lnms&tbm=isch&sa=X&ei=taUuVfadLOTlsQSu2ICABQ&ved=0CAYQ_A UoAQ#imgrc=_> Acesso em: 15 abr. 2015

Google Imagens. Disponível em: <https://www.google.com.br/search?q=imagens+*WAN*&espv=2&biw=1366&bih=643& source=lnms&tbm=isch&sa=X&ei=2Z8uVcrEFKvLsAT7ilCwCA&ved=0CAYQ_AUoA Q#tbm=isch&q=*LAN*+e+*WAN*+sem+fio> Acesso em: 15 abr. 2015

Google Imagens - Chave Assimétrica. Disponível em: <https://www.google.com.br/search?q=chaves+simetricas&espv=2&biw=1366&bih=643&source=lnms&tbm=isch&sa=X&ei=taUuVfadLOTlsQSu2ICABQ&ved=0CAYQ_A UoAQ#tbm=isch&q=chaves+assimetricas> Acesso em: 15 abr. 2015

Google Imagens - Chave Simétrica. Disponível em: <https://www.google.com.br/search?q=chaves+simetricas&espv=2&biw=1366&bih=643&source=lnms&tbm=isch&sa=X&ei=taUuVfadLOTlsQSu2ICABQ&ved=0CAYQ_AUoAQ#imgrc=_> Acesso em: 15 abr. 2015

Integrando a Base de Usuários *LDAP* com serviços de E-mail, *Proxy Web* e SSH. Disponível em: <https://www.google.com.br/url?sa=t&rct=j&q=&esrc=s&source=*web*&cd=1&cad=rja%2Findex.php%3Foption%3Dcom_phocadownload%26view%3Dcategory%26download%3D35%3A2008-2-integrando-a-base-de-usurios-ldap-com-servios-de-e-mail-proxy-*web*-e-ssh-lincon-ruam-angioletti-evaristo%26id%3D13%3Aredes-de-computadores%26Itemid%3D89&ei=zcl3VZ7AJcWbNqpD&usg=AFQjCNGVd4r6lu2lrwzG7C3J_rvSluiz40g&sig2=HlSO5yjx-2r1WSLfyv1Sg&bvm=bv.91071109,d.eXY> Acesso em: 22 abr. 2015

Introdução ao *cache* de *Web*. Disponível em: <https://memoria.rnp.br/newsgen/0003/*cache*.html> Acesso em: 22 abr. 2015

Introdução ao *Cache* de *Web*.. Claudia Watanabe, Rio de Janeiro, 2002 Disponível em: <www.rnp.br/newsgen/0003/*cache*.html> Acesso em: 9 set. 2014

LDAP. Disponível em: <http://www.hardware.com.br/termos/ldap> Acesso em: 22 abr. 2015

Mike Just - Criptografy III: Summetric Ciphers. Disponível em: <http://www.inf.ed.ac.uk/teaching/courses/cs/0910/lecs/symmetric-6up.pdf> Acesso em: 3 mar. 2015

Práticas de Segurança para Administradores de Redes Internet. Disponível em: <http://www.cert.br/docs/seg-adm-redes/seg-adm-redes.html> Acesso em: 22 abr. 2015

Proposta de Controle Eficaz do Acesso à Internet. Disponível em: <http://www.multicast.com.br/sergio/arquivos/monografia-pos-seguranca-proposta-de-controle-eficaz-do-acesso-a-internet.pdf> Acesso em: 22 abr. 2015

Proxy. Disponível em: <http://pt.wikipedia.org/wiki/**Proxy**> Acesso em: 17 fev. 2015

Redes sem fio no Mundo em Desenvolvimento: Um guia prático para o planejamento e a construção de uma infra-estrutura de telecomunicações. Disponível em: <http://pt.slideshare.net/jhribeiro/redes-sem-fio> Acesso em: 22 abr.2015

Segurança da Informação. Disponível em: <http://www.ice.edu.br/TNX/encontrocomputacao/artigos-internos/aluno_adrielle_fernanda_seguranca_da_informacao.pdf> Acesso em: 14 abr. 2015

Segurança em Redes sem Fio. Disponível em: <htps://pt.scribd.com/doc/133141005/Seguranca-em-redes-sem-fio> Acesso em: 14 out. 2014

Segurança em Redes sem Fio. Disponível em: <http://www.land.ufrj.br/~verissimo/cos871/bibref/wnsmono.pdf> Acesso em: 22 abr.2015

Squid. Disponível em: <http://wiki.ubuntu-br.org/**Squid**> Acesso em: 20 abr. 2015

Universidade Tecnológica Federal do Paraná. Disponível em: <http://www.utfpr.edu.br/a-instituicao> Acesso em: 22 abr. 2015

UTFPR-PG. Disponível em: <http://www.utfpr.edu.br/pontagrossa/o-campus> Acesso em: 22 abr. 2015