

**PROJETO FINAL DE GRADUAÇÃO EM
ENGENHARIA DE REDES DE COMUNICAÇÃO**

**IMPLEMENTAÇÃO DO NETWORK SIMULATOR 2
EM AMBIENTE WEB**

**Igor Carvalho Pereira
Leonardo Costa Santiago**

Brasília, julho de 2009

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**IMPLEMENTAÇÃO DO NETWORK SIMULATOR 2 EM
AMBIENTE WEB**

**Igor Carvalho Pereira
Leonardo Costa Santiago**

Orientador: Professor Georges Amvame-Nze

**PROJETO FINAL DE GRADUAÇÃO EM ENGENHARIA DE
REDES DE COMUNICAÇÃO**

PUBLICAÇÃO: 001/2009

BRASÍLIA / DF: JULHO/2009

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**IMPLEMENTAÇÃO DO NETWORK SIMULATOR 2
EM AMBIENTE WEB**

**IGOR CARVALHO PEREIRA
LEONARDO COSTA SANTIAGO**

PROJETO FINAL DE GRADUAÇÃO SUBMETIDO AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE ENGENHEIRO DE REDES DE COMUNICAÇÃO.

APROVADO POR:

**GEORGES AMVAME NZE, Doutor, UnB
(ORIENTADOR)**

**CARLOS HENRIQUE BACELLAR BON, Mestre, SERPRO
(EXAMINADOR EXTERNO)**

**FÁBIO LÚCIO LOPES DE MENDONÇA, Mestre, UnB
(EXAMINADOR EXTERNO)**

BRASÍLIA/DF, 02 DE JULHO DE 2009

FICHA CATALOGRÁFICA

PEREIRA, IGOR CARVALHO
SANTIAGO, LEONARDO COSTA

Implementação do Network Simulator 2 em ambiente WEB. [Distrito Federal] 2009.
(ENE/FT/UnB, Engenheiro de Redes de Comunicação, Engenharia de Redes de
Comunicação, 2009).

Projeto Final de Graduação - Universidade de Brasília, Faculdade de Tecnologia.
Departamento de Engenharia Elétrica.

1. NS
2. NAM
3. OTCL
4. Redes

REFERÊNCIA BIBLIOGRÁFICA

PEREIRA, IGOR CARVALHO, SANTIAGO, LEONARDO COSTA; 2009. Implementação do Network Simulator 2 em ambiente WEB. (Projeto Final de Graduação), Publicação Julho/2009, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF.

CESSÃO DE DIREITOS

NOMES DOS AUTORES: Igor Carvalho Pereira e Leonardo Costa Santiago

TÍTULO DO PROJETO: Implementação do Network Simulator 2 em ambiente WEB.

GRAU/ANO: Engenheiro de Redes de Comunicação/2009.

É concedida à Universidade de Brasília permissão para reproduzir cópias deste Projeto Final de Graduação e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. Os autores reservam outros direitos de publicação e nenhuma parte deste projeto final de graduação pode ser reproduzida sem a autorização por escrito dos autores.

Igor Carvalho Pereira

Leonardo Costa Santiago

DEDICATÓRIAS

Dedico este trabalho aos meus pais Jeremias e Flora que sempre acreditaram em meu potencial, à minha irmã Telma que sempre me orientou principalmente através do exemplo e à Livia pelo apoio constante em todos os projetos da minha vida.

Igor Carvalho Pereira

Dedico este trabalho à minha família, meus pais Isaias e Regina, meus irmãos Tiago e Flávio, minha avó Doraci, que me ajudaram e apoiaram para que o projeto fosse realizado.

Leonardo Costa Santiago

AGRADECIMENTOS

Ao nosso orientador Professor Doutor Georges Amvame-Nze, pelo constante apoio, incentivo, cobrança e amizade fundamentais para que este trabalho fosse realizado.

Aos bolsistas do Laboratório de Engenharia de Redes de Comunicação – LabRedes da Universidade de Brasília, por estarem sempre dispostos a ajudar quando foi solicitado.

Ao Wesley Gongora, que sempre nos informou sobre tudo o que foi preciso utilizar na elaboração do projeto, como modelos de configuração e outras monografias, por exemplo.

A todos os nossos sinceros agradecimentos.

Igor carvalho Pereira e Leonardo Costa Santiago

*"A alegria está na luta, na tentativa,
no sofrimento envolvido e não
na vitória propriamente dita."*

Mahatma Gandhi

RESUMO

As redes de computadores estão presentes em toda parte através de redes corporativas, redes domésticas, acesso sem fio a internet em áreas públicas, telefones celulares, rede de sensores, internet global e outros serviços em funcionamento além de novas aplicações em desenvolvimento. Devido ao alto custo e complexidade envolvidos na implantação de novos sistemas de comunicação, simuladores de redes tem um papel decisivo no projeto, análise e implementação de novas soluções em setores acadêmicos e privados. Este trabalho explora a possibilidade da utilização do Network Simulator (NS) remotamente através do desenvolvimento de uma ferramenta baseada em plataforma WEB que possibilita a interação do usuário com o simulador. Para usuários iniciantes o software permite a criação de simulações de uma rede LAN e WLAN em um ambiente interativo e para usuários avançados, é possível submeter arquivos NS2 com extensão tcl tendo todos os parâmetros requeridos para a simulação.

ABSTRACT

Computer networks are present everywhere through corporate networks, home networks, wireless Internet access in public areas, mobile phones, network sensors, global internet and other services in operation, in addition to new applications in development. Due to the high cost and complexity involved in the implementation of new communication systems, network simulators play a decisive role in the design, analysis and implementation of new solutions in academia and private sectors. This paper explores the possibility of using the Network Simulator (NS) remotely through the development of a WEB based platform that enables user interaction with the simulator. For beginners the software allows the creation of a LAN and WLAN network for simulations in an interactive environment, and for advanced users, it is possible to submit NS2 files with tcl extension having all the parameters required for the simulation.

ÍNDICE

1.	INTRODUÇÃO	15
1.1.	MOTIVAÇÃO	15
1.2.	OBJETIVO GERAL.....	15
2.	REDES DE COMPUTADORES	16
2.1.	REDES REAIS	16
2.1.1.	Internet.....	16
2.1.2.	HTTP e WEB.....	18
2.1.3.	Protocolo TCP/IP	20
2.1.4.	Redes de acesso.....	22
2.1.5.	Meios Físicos	24
2.1.6.	LANs	25
2.1.7.	MANs e WANs.....	26
2.2.	REDES VIRTUAIS.....	27
2.2.1.	Roteamento convencional.....	29
2.2.2.	LAN Emulation.....	30
2.2.3.	VPN.....	30
2.2.4.	VLAN.....	36
2.3.	REDES SEM FIO	39
2.3.2.	WLAN	40
3.	NETWORK SIMULATOR	44
3.1.	FUNCIONAMENTO DO NETWORK SIMULATOR	45
3.1.1.	Script TCL	45
3.1.2.	Declaração de Nós.....	45
3.1.3.	Declaração de Enlaces	46
3.1.4.	Cenário e Tráfego.....	46
3.2.	NAM - NETWORK ANIMATOR.....	47
4.	NETWORK SIMULATOR EM AMBIENTE WEB	48
4.1.	FUNCIONAMENTO DO SIMULADOR EM AMBIENTE WEB	50
4.1.1.	Início.....	50
4.1.2.	Simulação de LAN	51
4.1.3.	Simulação de WLAN	59

4.1.4.	Upload de arquivo tcl	61
4.1.5.	Projeto	65
4.1.6.	Links.....	66
5.	CONCLUSÃO	67
5.1.	TRABALHOS FUTUROS	68
6.	REFERÊNCIAS.....	70
7.	APÊNDICE A – ARQUIVO TCL	72

LISTA DE FIGURAS

Figura 2.1: Requisição das páginas WEB	20
Figura 2.2: Enlaces de acesso de sistemas finais a roteadores	22
Figura 2.3: Redes virtuais.....	28
Figura 2.4: Acesso remoto para VPN	31
Figura 2.5: Conexão de LAN com a Internet	31
Figura 2.6: Conexão de computadores em uma intranet.....	32
Figura 2.7: Processo de tunelamento	34
Figura 2.8: Formato de uma pequena VLAN.....	38
Figura 2.9: Redes WLAN.....	41
Figura 2.10: Acesso WLL	41
Figura 3.1: Tela do NAM com suas funcionalidades descritas	47
Figura 4.1: Arquitetura do projeto	49
Figura 4.2: Tela inicial da aplicação	51
Figura 4.3: Simulação de LAN.....	52
Figura 4.4: Exemplo de Simulação de LAN com parâmetros preenchidos.....	53
Figura 4.5: O arquivo de extensão tcl está pronto para ser submetido ao NS	55
Figura 4.6: Visualização do arquivo tcl gerado pela aplicação	56
Figura 4.7: Opções de visualizar o conteúdo do arquivo .nam ou assistir a animação no NAM.....	57
Figura 4.8: Aceitar a execução da simulação	58
Figura 4.9: NAM executando a animação.....	59
Figura 4.10: Simulação de WLAN.	62
Figura 4.11: Opção de upload de arquivo tcl.....	62
Figura 4.12: Enviando arquivo para upload.	63
Figura 4.13: Arquivo enviado com sucesso para o servidor.....	65
Figura 4.14: Detalhes sobre o projeto e instruções	66
Figura 4.15: Links relevantes ao projeto.....	67

LISTA DE ACRÔNIOS

CBR - *Constant Bit Rate*

CONSER - *Collaborative Simulation for Education and Research*

DARPA - *Defense Advanced Research Projects Agency*

EVC – *Ethernet Virtual Connection*

FTP – *File Transfer Protocol*

HTML – *HyperText Markup Language*

HTTP - *Hypertext Transfer Protocol*

ICIR - *The International Computer Science Institute Center for Internet Research*

IP - *Internet Protocol*

IPsec – *IP Security Tunnel Mode*

ISO - *International Standards Organization*

ISP – *Internet Service Provider*

L2F – *Layer 2 Forwarding*

L2TP – *Layer 2 Tunneling Protocol*

LAN – *Local Area Network*

MAC – *Media Access Control*

MAN – *Metropolitan Area Network*

MIT - *Massachusetts Institute of Technology*

NAM - *Network Animator*

NetBUEI – *NetBIOS Extended User Interface*

NS - *Network Simulator*

NS-2 - *Network Simulator Segunda Geração*

OSI - *Open Systems Interconnection*

PHP - *Hypertext Preprocessor*

PPTP – *Point to Point Tunneling Protocol*

PTP – *Point To Point*

QoS – *Quality of Service*

SAMAN - *Simulation Augmented by Measurement and Analysis for Networks*

TCP - *Transmission Control Protocol*

TLS – *Transport Layer Security*

UDP - *User Datagram Protoco*

VLAN – *Virtual Local Area Network*

VPN – *Virtual Private Network*

WAN – *Wide Area Network*

WEB – *World Wide Web*

WiMAX – *Worldwide Interoperability for Microwave Access*

WLAN – *Wireless Local Area Network*

WLL – *Wireless Local Loop*

WMAN – *Wireless Metropolitan Area Network*

WPAN – *Wireless Personal Area Network*

WWAN – *Wireless Wide Area Network*

1. INTRODUÇÃO

A utilização do mais popular simulador de redes do meio acadêmico é dificultada pela complexidade de sua instalação e utilização. Para construir um cenário de simulação, é necessário a utilização da linguagem interpretada OTCL desenvolvida pelo MIT (Massachusetts Institute of Technology). Com o objetivo de tornar mais amigável o primeiro contato de estudantes com o Network Simulator 2, este trabalho propõe interações com o simulador através de cenários de simulação construídos no ambiente WEB.

1.1.MOTIVAÇÃO

Apresentar em software livre uma ferramenta WEB que possa interagir com o Network Simulator 2. Dessa forma, alunos interessados em algum tipo de pesquisa, pesquisadores e acadêmicos podem observar e verificar como determinada rede está funcionando e se comportando, podendo fazer isso remotamente. Além da praticidade e do conforto de poder ver a rede de qualquer lugar, desde que possua acesso à Internet, não é preciso instalar o software na própria máquina, pois está tudo em um servidor, economizando recursos.

1.2.OBJETIVO GERAL

Desenvolver software para ambiente WEB utilizando a linguagem PHP com a finalidade de gerar arquivos na linguagem OTCL que possam ser interpretados pelo simulador de redes Network Simulator 2 e permitir que usuários façam *upload* de seus próprios arquivos de extensão TCL para utilizar o NS2 remotamente.

2. REDES DE COMPUTADORES

Uma rede de computadores consiste de 2 ou mais computadores e outros dispositivos conectados entre si de modo a poderem compartilhar seus serviços, que podem ser: dados, impressoras, mensagens (e-mails), etc.

2.1. REDES REAIS

Para se iniciar um estudo sobre as redes de computadores, é importante citar a Internet, que acaba por impulsionar o desenvolvimento das redes. O livro *Redes de computadores e a Internet* [8] foi usado como referência para se falar sobre a Internet, o HTTP e a WEB, o protocolo TCP/IP, as redes de acesso, os meios físicos e as LANs.

2.1.1. Internet

A Internet é uma rede pública e de livre acesso, ela é uma rede global. Existem também redes privadas, como redes corporativas e governamentais, que impedem a troca de mensagens entre hospedeiros de dentro e de fora de tais redes. Elas são chamadas de intranets, já que utilizam os mesmos roteadores, hospedeiros, enlaces e protocolos da rede global. Pode-se citar cinco camadas da internet: aplicação, transporte, rede, enlace e física.

Na camada de aplicação ocorre a interface entre o protocolo de comunicação e o aplicativo que solicitou ou que vai receber a informação através da rede.

Na camada de transporte destacam-se a comunicação confiável por uma camada de rede não confiável, o estabelecimento ou encerramento de conexões, o controle de congestionamento e de fluxo.

Na camada de rede tem-se a determinação dos melhores caminhos a serem percorridos, pelos pacotes, entre dois roteadores e a interconexão entre várias redes heterogêneas.

Na camada de enlace são detectados e corrigidos os erros que possam acontecer na camada física. Aqui também ocorre a transmissão e a recepção de quadros, além do controle de fluxo.

A camada física faz a definição das características técnicas dos dispositivos físicos do sistema. Contém equipamentos de cabeamento ou outros tipos de canais de comunicação que se comunicam diretamente com o controlador da interface de rede. Pode realizar um controle de erros básicos.

Uma rede de computadores interconecta milhões de equipamentos de computação em todo o mundo. Esses equipamentos são computadores, desktops ou notebooks, servidores que armazenam e transmitem informações, como mensagens de email, por exemplo. Porém quando se fala em redes de comunicação, pode-se citar muitos outros dispositivos, como celulares, PDAs ou TVs, todos esses equipamentos são chamados de sistemas finais.

Enlaces de comunicação interligam os diferentes sistemas finais, sendo que cada enlace é constituído por diferentes tipos de meios físicos, cabos coaxiais, fibras óticas, fios de cobre. Além disso, cada um tem sua própria taxa de transmissão que é dada em bits por segundo.

Como existem vários enlaces conectados, para que a informação, conhecida como pacote, chegue ao seu destino, são necessários comutadores de pacotes, que realizam justamente a função de transmitir os pacotes através da rede. A informação chega no enlace de entrada, é lida e passada para o enlace de saída, sendo enviada ao próximo comutador. Os comutadores mais utilizados são os roteadores e os *switchs*.

A sequência de enlaces percorridos pelo pacote, do remetente até o emissor, é chamado de caminho ou rota. Existem duas maneiras de se fazer esse caminho usando comutação de pacotes ou comutação de circuitos. Na comutação de pacotes, o remetente envia os pacotes sem que seja estabelecida uma rota fixa, de acordo com que eles vão chegando aos enlaces de entrada dos comutadores, vão sendo enfileirados, aguardando que sejam enviados ao seu destino, e isso pode causar o atraso de fila. Na comutação de circuitos, é estabelecido um canal antes que os dados sejam enviados, e após isso, começasse a transmissão. Isso causa um atraso inicial e gasta muito recurso da rede.

Para que se controle o envio e o recebimento de informações, é necessária a execução de protocolos. Na internet, por exemplo, têm-se dois protocolos mais importantes, o TCP (*Transmission Control Protocol*) e o IP (*Internet Protocol*).

2.1.2. HTTP e WEB

Até 1990, a Internet era utilizada, principalmente, por pesquisadores, professores e estudantes universitários para transferir arquivos de hospedeiros locais para remotos e vice-versa, enviar e receber notícias e correio eletrônico. Sendo assim, a Internet não era muito conhecida fora de comunidades acadêmicas e de pesquisa. Isso até o surgimento da Web, que possibilita o envio e recebimento do que se quer e a hora que se deseja, diferentemente do rádio ou da televisão. Além disso, é fácil disponibilizar informações na Web, hiperlinks e buscadores ajudam quando se navega nos sites Web, formulários, e alguns dispositivos permitem a interação com páginas e sites e cada vez mais se oferecem menus de interfaces para vários vídeos e áudios armazenados na internet, podendo ser acessados por demanda.

O HTTP (*HyperText Transfer Protocol*) é um protocolo de camada de aplicação da Web. Ele é implementado em dois programas: um programa servidor e outro cliente. Os dois programas, que são executados em diferentes sistemas finais, se comunicam por meio de troca de mensagens HTTP.

Uma página Web é formada por objetos, que são arquivos simples, como uma imagem JPEG, por exemplo. Esses objetos podem ser acessados por uma única URL. A grande maioria das páginas Web é constituída por um arquivo-base HTML e outros muitos objetos referenciados.

Um browser é um agente de usuário Web. Ele mostra a página solicitada e fornece várias características de navegação e configuração. Netscape e Internet Explorer são exemplos de browsers bastante utilizados.

Um servidor Web abriga objetos Web, cada um é endereçado por um URL. Servidores Web fazem a implementação do lado servidor do HTTP e entre os mais conhecidos estão o Microsoft Internet Information Server e o Apache.

O HTTP faz a definição de como os clientes Web farão a requisição das páginas aos servidores e como esses servidores farão a transferência aos clientes. A figura abaixo mostra como a requisição é feita.

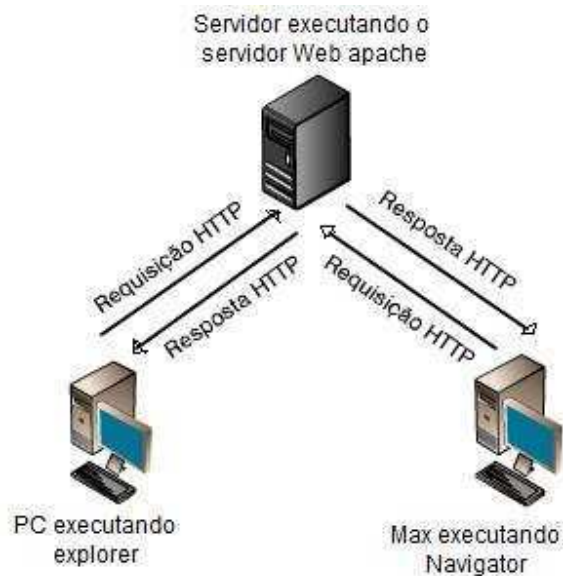


Figura 2.1: Requisição das páginas WEB

O usuário requisita uma página Web e o browser envia a mensagem de requisição para os objetos da página. O servidor recebe e responde a requisição que contém os objetos

2.1.3. Protocolo TCP/IP

Além de utilizar o livro de James F. Kurose e Keith W. Ross [8], para se falar do protocolo TCP/IP, utilizou-se o livro de Douglas E. Comer [9]. Dessa forma, pode-se dizer que redes TCP/IP oferecem dois tipos de serviços à aplicações de sistemas finais, os serviços orientados para conexão e os serviços não orientados para conexão.

No primeiro serviço, os programas de cliente e servidor enviam pacotes de controle um para o outro antes de enviar pacotes com as informações que serão transmitidas. Isso ocorre para que cliente e servidor se preparem para a transferência de pacotes estabelecendo uma conexão entre

os dois sistemas finais. Os serviços orientados para conexão possuem outros serviços como transferência confiável de dados, controle de fluxo e controle de congestionamento.

A transferência confiável de dados diz que pode-se confiar que a conexão vai entregar os dados na ordem correta e sem erros. Essa confiabilidade é obtida utilizando confirmações e retransmissões.

O controle de fluxo impede que um dos lados da conexão envie pacotes muito rapidamente, impedindo a sobrecarga entre eles. Esse controle é feito com a implementação de buffers de envio e de recebimento nos sistemas finais.

O controle de congestionamento ajuda a evitar o travamento da Internet. Se um comutador de pacotes ficar cheio, haverá grandes filas de pacotes e pode haver perdas de algum deles. Caso continuem os envios de dados, vai ocorrer um travamento e poucos pacotes, em tempos longos, chegarão aos seus destinos. Os sistemas finais sabem que há congestionamento quando param de receber confirmações dos pacotes que foram enviados.

É importante saber que um serviço é orientado para conexão mesmo que não possua alguma das três características citadas, pois qualquer protocolo que faça apresentação entre as entidades que estão se comunicando antes de transferir os dados, é um serviço orientado. O TCP é um exemplo desse tipo de serviço.

No serviço não orientado para conexão, não ocorre apresentação mútua para que seja estabelecida a conexão. Se um lado de uma aplicação deseja enviar pacotes à outro lado, eles simplesmente são transmitidos. Isso pode ocasionar uma entrega de dados mais rápida, mas

também não há confiança na transferência das informações, nem controle de fluxo e nem controle de congestionamento, é o UDP (*User Protocol Datagram*).

Algumas aplicações que utilizam o TCP são: Telnet (para login remoto), FTP (para transferência de arquivos) e HTTP (para WEB). Outras aplicações utilizam o UDP como protocolo, como telefone por Internet e videoconferência, por exemplo.

2.1.4. Redes de acesso

Redes de acesso são os enlaces físicos que fazem a conexão de um sistema final com o primeiro roteador de uma rota entre dois sistemas finais, chamado roteador de borda. A figura abaixo mostra enlaces de acesso de sistemas finais a roteadores de borda.

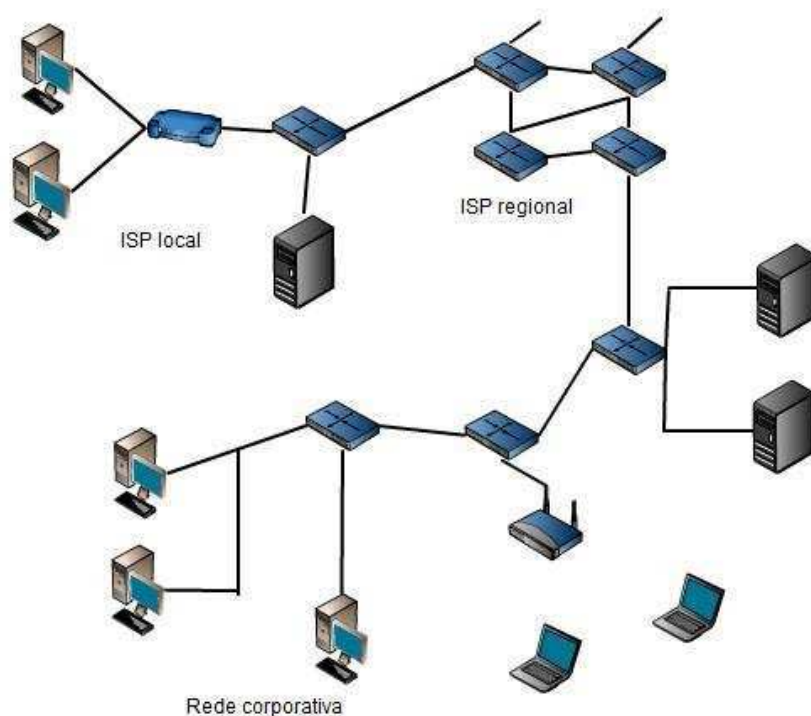


Figura 2.2: Enlaces de acesso de sistemas finais a roteadores

Existem três categorias de redes de acesso, acesso residencial, acesso corporativo e acesso sem fio.

O acesso residencial liga sistemas finais residenciais a um roteador de borda. Uma forma de acesso é o modem discado ligado a um ISP residencial por uma linha telefônica. Nesse caso a conexão é ponto a ponto discada. Porém hoje em dia, o uso de acesso banda larga vem aumentando, dessa forma pode-se ter taxas de transferência de bits bem mais altas e pode-se utilizar o telefone, independente do uso da Internet. Outro aspecto importante da Internet banda larga é que se tem acesso permanente a conexão.

O acesso corporativo se aplica a empresas e campus universitários. Geralmente utiliza-se uma rede local (LAN) para fazer a ligação de um sistema final ao roteador de borda. A Ethernet é a tecnologia mais usada nessas redes, usualmente com velocidades de 100 Mbps, 1 Gbps e até 10 Gbps. É usado par trançado ou cabo coaxial para se estabelecer as conexões entre os vários sistemas finais.

Com relação ao acesso sem fio, pode-se dizer que existem duas categorias que são muito utilizadas: as LANs sem fio, em que os usuários trocam pacotes com uma estação base (ponto de acesso sem fio) dentro de um raio de alcance de alguns metros e as redes sem fio de acesso de longa distância, em que um provedor de telecomunicações gerencia a estação base e atende usuários dentro de um raio de quilômetros. Para as LANs sem fio, usa-se muito a tecnologia IEEE 802.11, que possui uma velocidade de até 54 Mbps.

2.1.5. Meios Físicos

Os bits, quando transmitidos, passam por uma série de pares transmissores-receptores, estes por sua vez, os recebem por meio de ondas eletromagnéticas ou pulsos óticos, Essas ondas ou pulsos, se propagam por um meio físico. Os meios físicos são divididos em duas categorias: meios guiados e meios não guiados. Nos meios guiados, as ondas passam por um meio sólido, como fibra ótica, por exemplo. Nos meios não guiados, as ondas propagam-se na atmosfera, como as LANs, por exemplo.

O par trançado é o meio de transmissão mais barato e mais comum que usa. O par é formado por dois fios de cobre isolados e enrolados em espiral. Cada par de fios constitui apenas um enlace de comunicação. As taxas de transmissão dependem do fio e da distância entre emissor e receptor.

O cabo coaxial também é formado por dois fios de cobre, mas concêntricos ao invés de paralelos. Dessa forma, pode alcançar altas taxas de transmissão de bits. Cabos coaxiais são bastante utilizados em sistemas de TV a cabo.

A fibra ótica conduz pulsos de luz e cada pulso representa um bit. Cada fibra pode suportar vários pulsos luminosos, ou seja, possui elevadas taxas de transmissão. Fibras óticas são muito utilizadas em transmissões a longa distância porque são imunes à interferência eletromagnética e têm baixa atenuação de sinal. O seu maior problema é que possui custos muito elevados.

Canais de rádio carregam sinais dentro do espectro eletromagnético, sendo assim, não requerem cabos e dão conectividade a usuários móveis.

2.1.6. LANs

Além da referência [8], também pesquisou-se sobre as LANs em [16], fazendo-se algumas correções e atualizações de informações. Redes de áreas locais (LANs) são redes de computadores usadas com a finalidade de troca de dados. LAN é um conjunto de equipamentos, hardwares e softwares, que permitem o estabelecimento de ligação entre diferentes computadores, assim pode-se trocar ou compartilhar arquivos e outras informações. Essas redes são locais porque cobrem pequenas distâncias, cerca de 10 km, no máximo.

As LANs são usadas para conectar estações que estejam em casa, escritório, em escolas e alguns prédios próximos.

Em uma LAN temos: servidores, clientes, sistema operacional de rede, meios de transporte, dispositivos de rede e protocolos de comunicação.

Os servidores são computadores com elevada capacidade de processamento e armazenamento e tem a função de disponibilizar arquivos, serviços ou aplicações na rede existente. Alguns exemplos de serviços fornecidos são email e firewall, por exemplo. Como servidor de arquivos pode armazenar dados que sejam importantes e confidenciais. Já como servidor de aplicações possui a aplicação instalada e a disponibiliza aos computadores que estejam conectados à rede.

Os clientes são estações de trabalho, desktops ou notebooks, que possuem acesso aos serviços ou aplicações disponibilizados pelo servidor. Possuem processamento e armazenamento menores que dos servidores.

Sistema operacional de rede é um programa que controla o computador. Faz com que os programas e funções do computador possam ser acessados pelo usuário, por meio de uma

interface gráfica. Os mais utilizados são o sistema UNIX, que é cliente e servidor, e o Windows, que possui versões para servidor e outras para clientes.

Os meios de transporte mais usados são a ethernet e o wireless, que são transmitidos por cabos e pelo ar, respectivamente.

Os dispositivos de rede são os meios físicos que realizam a comunicação entre os equipamentos da rede. Por exemplo, roteadores e *switchs*.

Protocolos de comunicação são as linguagens utilizadas para comunicação entre os dispositivos da rede, é necessário que exista uma linguagem padrão. Os protocolos mais usados são o TCP/IP, o NetBEUI e o UDP.

2.1.7. MANs e WANs

As MANs e WANs apresentam um funcionamento bem parecido com o das LANs, então só foi alguns detalhes foram fornecidos, tendo como base a referência [16] e fazendo correções, tendo por base informações obtidas na disciplina Redes de comunicação 1, ministrada pelo professor Paulo Gondim.

MAN – *Metropolitan Area Network* – é o nome das redes que ocupam o perímetro de uma cidade ou região metropolitana. Permitem que empresas que tenham filiais em diferentes bairros se conectem entre si. Geralmente é maior que a LAN e menor que a WAN.

A WAN – *Wide Area Network* – é uma rede de longa distância que abrange uma grande região geográfica, normalmente um país ou um continente.

O tráfego das WANs aumenta cada vez mais com o passar do tempo, fazendo com que haja o surgimento de congestionamentos cada vez maiores de dados que são transportados pela rede. O tráfego da rede deve ser modelado por meio de medições com um certo grau de resolução, que inclui a análise de pacotes, assim é possível obter estatísticas gráficas e descritivas sobre o que ocorre na rede. Caso ocorra variação na chegada dos pacotes, verifica-se que a WAN está consistente e que seu tráfego pode ser acelerado para cada tipo de serviço, como voz ou vídeo, por exemplo.

A qualidade de serviço de uma WAN é definida pelo QoS – *Quality of Service* – que acaba sendo determinado pelo tráfego em algumas tecnologias como o IP e o ATM, entre outros. A qualidade de serviço é a capacidade que a rede possui de transmitir os dados de maneira consistente e previsível.

O QoS pode ser utilizado nos seguintes recursos: classificação de pacotes, gerenciamento de banda e controle de admissão, prevenção de congestionamento e medição de serviços e tráfego com granularidade.

Alguns protocolos possibilitam a transmissão de dados de uma WAN, o PPP, o X.25, o *Frame Relay*, as redes ATM e a DSL.

2.2. REDES VIRTUAIS

Apesar de não terem sido usadas no trabalho em si, é importante falar sobre redes virtuais e sua utilidade. Mesmo porque futuramente pode ser interessante se visualizar essas redes em algum simulador, via web.

Redes virtuais são grupos de estações e servidores que se comunicam, independentemente, de sua topologia ou de sua localização física, como se fosse uma mesma rede lógica.

Redes virtuais podem ser criadas, por exemplo, tendo por base grupos de endereço MAC, LAN *emulation* ou grupos de portas de um determinado switch.

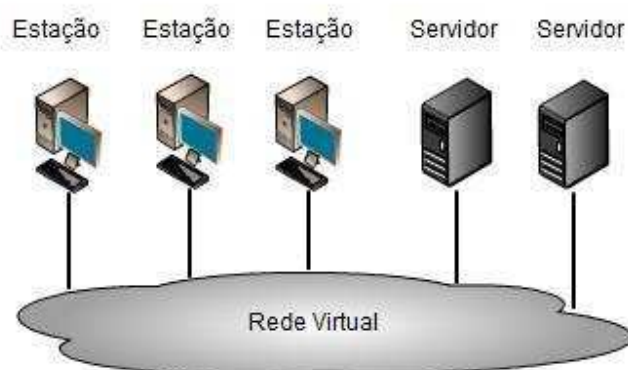


Figura 2.3: Redes virtuais

Uma rede virtual consiste, em partes, de links de redes virtuais, que não são conexões físicas (wireless ou cabeadas) entre diferentes máquinas, mas sim implementações que usam métodos de virtualização de redes.

As duas formas mais comuns de redes virtuais são protocolos base de redes virtuais, como VLAN ou VPN, e redes virtuais baseadas em dispositivos virtuais, como máquinas virtuais, por exemplo. As duas formas podem ser utilizadas.

VLANs são criadas são criadas pelo particionamento de uma LAN em várias subredes. As redes divididas podem estar em um roteador somente, podem estar em vários roteadores ou podem estar em uma VPN.

Uma VPN consiste em múltiplos pontos finais remotos (tipicamente roteadores), unidos por algum tipo de túnel em detrimento de outra rede, geralmente uma terceira rede. Dois pontos finais constituem uma rede virtual privada ponto a ponto, PTP VPN. Conectando mais que dois pontos finais com a rede de túneis, cria-se um VPN multipontos.

Um VPLS é um tipo específico de VPN multiponto. São divididos em Serviços de LAN transparentes (TLS) e em serviços de conexão virtual Ethernet (EVCS). A diferença entre eles é que um TLS envia o que recebe, enquanto um EVCS adiciona um VLAN ID, ou seja, faz separação geográfica e de VLANs. Veremos a seguir alguns métodos de padronização para se criar redes virtuais.

2.2.1. Roteamento convencional

Para se trabalhar com redes que utilizem comutação de pacotes, Fast Ethernet ou Ethernet, por exemplo, a única maneira padronizada é a criação de sub-redes lógicas que utilizem roteamento IP ou IPX. Como o roteador vai criar domínios de broadcast diferentes, cada empresa vai definir suas regras de acesso de acordo com suas necessidades.

2.2.2. LAN Emulation

Esse serviço permite que o maior número possível de aplicações usadas atualmente possa usar o ATM sem que sejam necessárias modificações, ao mesmo tempo em que as estações ligadas em outras redes possam se comunicar com estações ligadas em ATM.

O LAN *Emulation* – LANE – é implementado por meio de uma camada de software em estações que possuam interface ATM, independente do que seja, um switch, um roteador ou um host. Tem como função principal fornecer ao protocolo camada rede uma interface igual a que é oferecida por uma rede tradicional. Além dessa função, o LAN *Emulation* define o *bridging* entre os protocolos de uma rede e o ATM, permitindo a comunicação entre as estações ligadas a elas.

2.2.3. VPN

Para falar sobre VPN, baseou-se no artigo de Liou Kuo Chin [14], podendo-se dizer que *Virtual Private Network* – VPN – é uma rede de comunicações privadas geralmente usadas por empresas ou instituições, formada dentro de uma rede pública, como a Internet.

A segurança das VPNs vêm do uso da criptografia por tunelamento que garante confidencialidade, autenticação e integridade que permitem a privacidade das comunicações.

Existem três aplicações que se destacam para as VPNs: o acesso remoto via Internet, a conexão de LANs via Internet e a conexão de computadores em uma intranet.

O acesso remoto a redes de corporações, por meio da Internet, é feito com a VPN com uso da ligação local a algum provedor de acesso. O desktop disca para o provedor e se conecta à

internet e ao software de VPN criando uma rede virtual privada entre o usuário e o servidor de VPN.

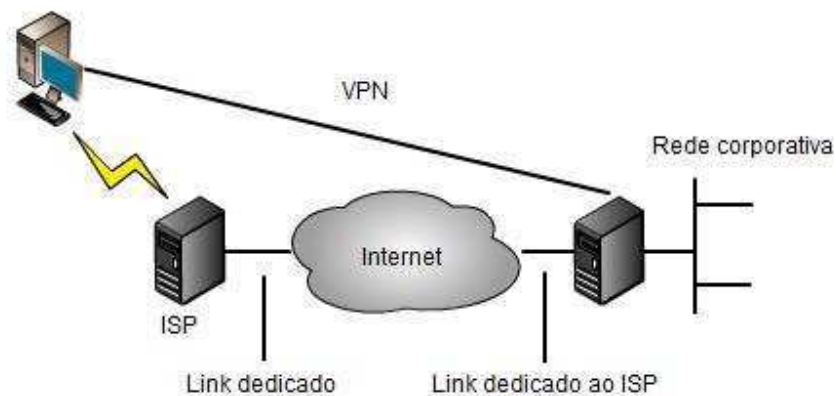


Figura 2.4: Acesso remoto para VPN

A conexão de LANs pela internet substitui as conexões entre LANs por meio de circuitos dedicados de longa distância. A VPN garante a interconexão formando uma WAN corporativa.

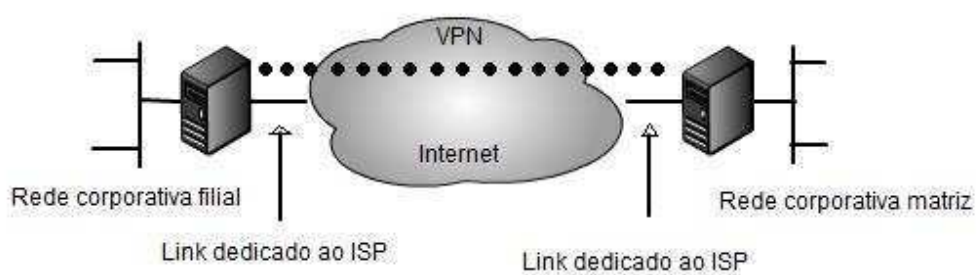


Figura 2.5: Conexão de LAN com a Internet

Em algumas corporações existem dados confidenciais que apenas algumas pessoas podem ter acesso. A conexão de computadores em uma intranet é uma solução para garantir essa confidencialidade das informações, mas acaba por criar dificuldades de acesso a dados que estão na rede da corporação por parte de cada departamento.

Usando a VPN o administrador da rede pode especificar quais usuários terão acesso aos recursos disponíveis na rede corporativa, pois a VPN possibilita a conexão física entre redes locais restringindo acesso por meio de um servidor VPN entre elas. Além da possibilidade de restringir acesso à rede, pode-se também criptografar a comunicação ao longo da VPN, garantindo a confidencialidade dos dados.

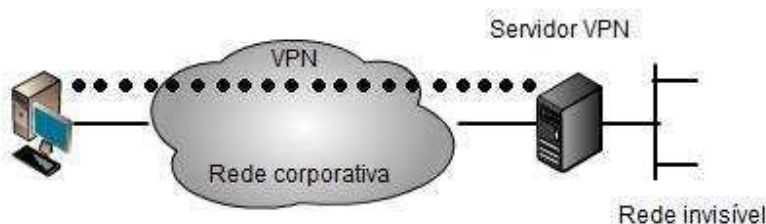


Figura 2.6: Conexão de computadores em uma intranet

Para se estabelecer uma VPN são necessários alguns requisitos básicos. Ela deve dispor de recursos que permitam o acesso de usuários remotos, que tenham acesso permitido aos recursos da LAN corporativa, viabilizem a interconexão das LANs, possibilitando assim o acesso de filiais, compartilhando informações e assegurem a privacidade e integridade dos dados ao passar pela internet. Os requisitos são: autenticação de usuários, gerenciamento de endereço, criptografia de dados, gerenciamento de chaves e suporte a múltiplos protocolos.

A autenticação de usuários verifica a identidade do usuário para ver se ele possui autorização de acesso. Deve haver disponibilidade de formas de auditoria para prover informações sobre os acessos feitos, como quem acessou, o que foi acessado e quando esse acesso foi realizado.

O gerenciamento de endereço cuida para que o endereço do usuário em sua rede privada não seja divulgado, adotando endereços 'falsos' para o tráfego externo.

A criptografia de dados garante a segurança dos dados. Esses trafegam na rede pública e na rede privada e caso sejam interceptados por alguém que não tenham autorização para ver essas informações, não poderão ser visualizados pois os dados estão codificados.

A utilização de chaves garante a segurança das mensagens que estão criptografadas e funciona como um segredo que apenas as partes envolvidas compartilham. O gerenciamento de chaves faz com que a troca periódica dessas mensagens seja garantida, possibilitando assim uma comunicação segura.

Como existem diversos protocolos que podem ser utilizados o VPN suporta vários deles, pelo menos os principais, como o IP ou o IPX, por exemplo.

As redes virtuais privadas são baseadas no tunelamento, que pode ser definido como um processo em que se encapsula um processo dentro de outro. Antes que o encapsulamento ocorra, o pacote que vai ser transportado é criptografado para ficar ilegível caso ocorra algum tipo de interceptação durante o seu transporte. Após ser criptografado e encapsulado, o pacote passa pela internet até seu destino, onde ocorre o desencapsulamento e a decriptografia. É importante ressaltar que pacotes de um determinado protocolo podem ser encapsulados em pacotes de outros

protocolos como, por exemplo, pacotes IPX podem ser encapsulados e até transportados dentro de pacotes TCP/IP.

Dessa forma verifica-se que o tunelamento possui o encapsulamento, a transmissão ao longo da rede e o desencapsulamento em seu processo como um todo.

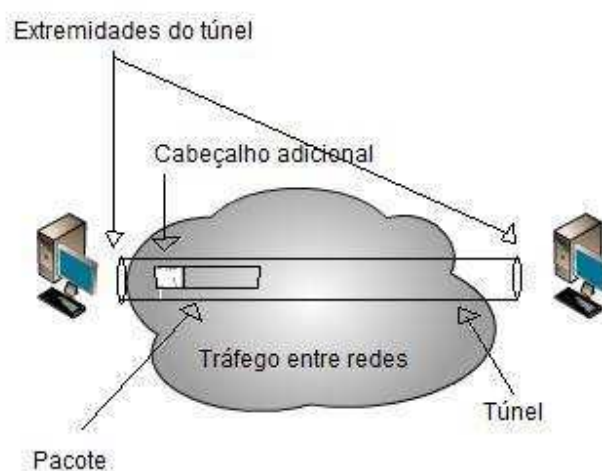


Figura 2.7: Processo de tunelamento

Para que se estabeleça um túnel é preciso que seja utilizado o mesmo protocolo de tunelamento nas suas extremidades.

Caso o tunelamento ocorra na camada 2 do modelo OSI, ou seja, a camada de enlace, terá como objetivo transportar protocolos de nível 3, como o IP e o IPX, por exemplo. Os protocolos vão usar quadros como unidade de troca, encapsulando os pacotes da camada 3 em quadros PPP, *Point to Point Protocol*. Pode-se citar, por exemplo, o PPTP, o L2TP e o L2F.

O PPTP – *Point to Point Tunneling Protocol* – possibilita a criptografia e encapsulamento do tráfego IP, IPX e NetBEUI, assim podem ser enviados por redes IP privadas ou públicas.

O L2TP – *Layer 2 Tunneling Protocol* – possibilita que o tráfego IP, IPX e Net BEUI possam ser criptografados e enviados por meio de canais datagrama ponto a ponto, como x25, IP ou ATM, por exemplo.

O L2F – *Layer 2 Forwarding* – é usado para VPN discada.

Quando ocorre tunelamento em nível 3, camada de rede, são encapsulados pacotes IP com um cabeçalho do mesmo protocolo, depois disso são enviados pela rede.

O IPsec – *IP Security Tunnel Mode* – possibilita que pacotes IP recebam cabeçalho adicional desse mesmo protocolo durante a criptografia e o encapsulamento, então os pacotes podem ser transportados em uma rede pública ou privada.

No tunelamento em nível 2, o túnel dito corresponde a uma sessão, em que as extremidades do túnel fazem negociação de configuração dos parâmetros que farão o estabelecimento do túnel, entre essas configurações estão o endereçamento, a criptografia e os parâmetros de compressão.

O gerenciamento do túnel é feito por meio de protocolos de manutenção e nesses casos necessita-se que o túnel seja criado, mantido e finalizado.

No tunelamento em nível 3, ocorre o mesmo que na tecnologia anterior, com exceção da fase de manutenção do túnel, inexistente em nível de camada de rede.

Depois de se estabelecer o túnel, é iniciado o envio dos dados. O cliente ou o servidor do túnel usa um protocolo de tunelamento e prepara o pacote para o transporte, então o cliente manda o

pacote, já encapsulado, na rede que vai roteá-lo até o servidor do túnel. O servidor vai receber o pacote, desencapsulá-lo e encaminhá-lo à rede destino.

2.2.4. VLAN

Utilizando informações escritas por Frederico Madeira, em um artigo publicado na Internet [15], pode-se dizer que uma VLAN – *Virtual Location Area Network* – é uma rede logicamente independente. Podem existir várias VLANs em um mesmo switch.

Inicialmente, as VLANs eram configuradas para diminuir o tamanho do domínio de colisão, que é uma área lógica em que os pacotes podem colidir uns contra os outros, em uma rede ethernet muito grande, assim melhorando o seu desempenho. Assim que os *switchs* começaram a eliminar esse problema iniciou-se um processo para diminuir o domínio de broadcast, que é a capacidade que um computador tem de ser conectado a outro dispositivo da rede sem a necessidade de usar um roteador, na camada MAC. Outro motivo para se ter uma VLAN é de restringir acesso a recursos disponíveis na rede.

As VLANs podem ser configuradas de diferentes maneiras: em nível do protocolo IP ou IPX, por exemplo; baseada no endereço MAC; baseada na sub-rede IP ou baseada na porta, ou seja, no mundo real.

As VLANs podem ser estáticas ou dinâmicas. As estáticas são baseadas em portas, isto é, qualquer dispositivo conectado a uma determinada porta de um switch pertence a uma certa VLAN. A VLAN dinâmica é baseada no endereço MAC, então o administrador da rede deve cadastrar, previamente, os endereços MAC das estações de trabalho e associá-los a suas VLANs. Dessa forma, assim que o usuário plugue seu computador na rede, ele estará na VLAN certa.

Pode-se fazer o estabelecimento de uma VLAN de duas maneiras: por marcação de quadro – *frame-tagging* – e por filtragem de quadro – *frame-filtering*.

A marcação de quadro muda a informação que está dentro do quadro da camada 2, de tal forma que os *switchs* podem encaminhar o tráfego da VLAN para suas VLANs de destino, em seguida, voltar o quadro para seu formato normal.

A filtragem de quadro faz com que o switch procure por um critério no quadro da camada de enlace e use esse sistema de comparação para que se encaminhe o tráfego para a VLAN certa.

Existem três formas de se implementar uma VLAN para um dispositivo de camada de enlace:

VLANs abertas, que possuem apenas um banco de dados de endereço MAC para todas as VLANs.

VLANs fechadas, que possuem bancos de dados de endereço MAC separados para cada VLAN.

VLANs de modo mixado cada VLAN pode ser configuradas das duas maneiras, aberta ou fechada.

Dois protocolos são muito utilizados na configuração de VLANs, eles são usados em troncos de ligação entre diferentes *switchs*.

O ISL – *InterSwitch Link* - é usado em um enlace tronco. Se o switch estiver configurado para usar esse protocolo, ele irá encapsular os pacotes que saem pela interface, de maneira que qualquer switch conectado a essa interface e que esteja configurado com o ISL pode desencapsular esses pacotes e encaminhá-los para suas VLANs de destino. Esse protocolo

adiciona um cabeçalho, conhecido como VLAN ID, ao pacote, que é preenchido pelo ID da VLAN que o quadro pertence.

O protocolo 802.1Q é padrão do IEEE. É um protocolo aberto e tem um funcionamento muito parecido com o funcionamento do protocolo ISL. Também adiciona um cabeçalho ao pacote, mas não há necessidade do protocolo estar rodando para desencapsular os pacotes.

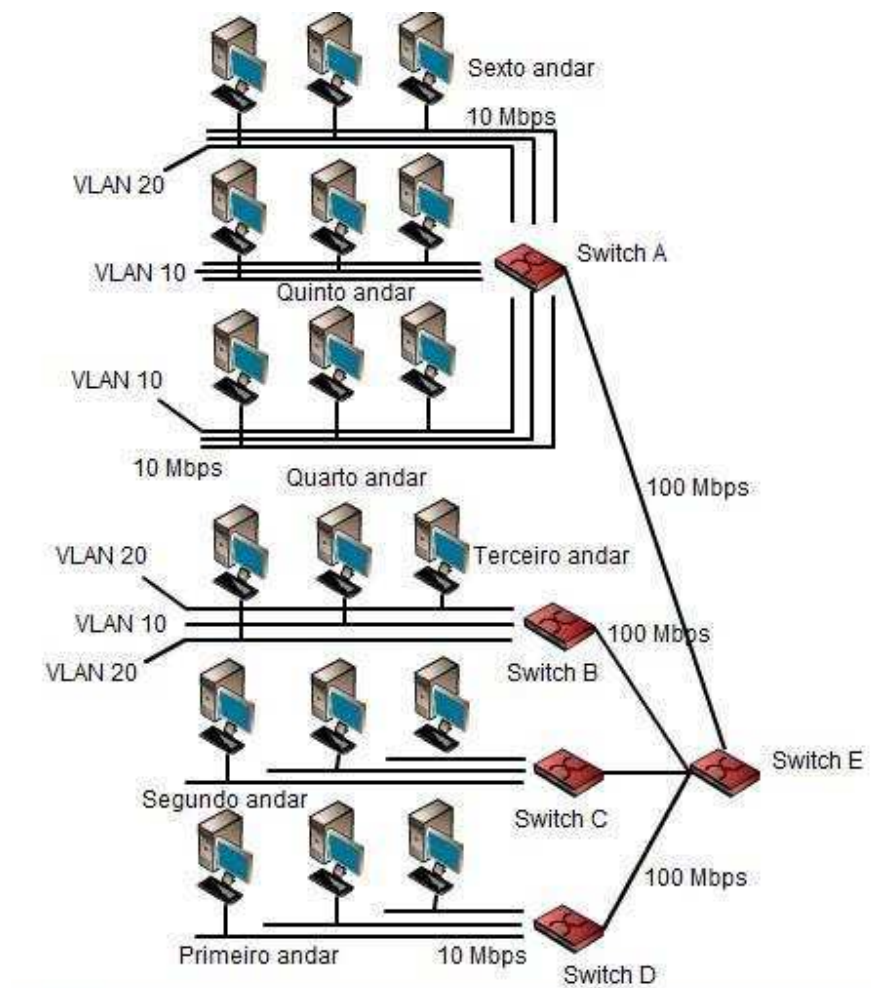


Figura 2.8: Formato de uma pequena VLAN

2.3.REDES SEM FIO

Uma rede sem fio ou rede wireless é uma rede que não necessita do uso de cabos, sejam eles de qualquer tipo. Os equipamentos utilizam radiofrequência ou infravermelho para estabelecer a comunicação.

O uso mais comum dessa tecnologia é em redes de computadores e serve para prover acesso à Internet em locais como escritório, parque, em casa e até mesmo no próprio local de trabalho – empresa.

Existem vários tipos de redes sem fio: WPAN, WLAN, WMAN, WWAN e WLL.

2.3.1. WPAN

Com base no artigo encontrado na Internet [18], podemos dizer que as WPANs – *Wireless Personal Area Network* – são redes sem fio que possuem um pequeno alcance, apenas algumas dezenas de metros, sendo geralmente utilizadas para ligar alguns periféricos como, impressoras ou telefones portáteis, por exemplo. Pode ser usado também para estabelecer comunicação entre dois computadores que estejam bem próximos.

A principal tecnologia WPAN é o *Bluetooth*, que gasta pouquíssima energia e é bastante adaptado à utilização em pequenos periféricos. Possui um alcance de, aproximadamente, 30 metros.

Outra tecnologia WPAN é o *ZigBee* que permite a obtenção de ligações sem fio com baixo consumo de energia. Essa tecnologia é eficaz sobre a banda de frequências dos 2,4 GHz, permitindo a obtenção de débitos que podem chegar a 250 Kb/s.

2.3.2. WLAN

Para falar de WLAN, WMAN e WWAN, foram utilizados como referência, um trabalho final de graduação [20] e algumas informações do Wikipédia [16]. Além disso, para falar um pouco mais sobre WLAN, usou-se o artigo de Adailton J. S. Silva [17].

As WLANs – *Wireless Local Area Network* – fornecem as mesmas funcionalidades das redes com fio, porém de maneira flexível, de fácil configuração e com uma boa conectividade em áreas de prédios ou de algum campus. As WLANs podem atingir distâncias de até 180 metros, isso dependendo da tecnologia, da rádio frequência ou do infravermelho utilizados.

Dessa forma ocorre uma combinação de mobilidade com altas taxas de conexão, em alguns casos.

A comunicação de dados entre os pontos da rede se dá por meio do uso de portadoras de rádio ou infravermelho, que modulam a informação e transmitem através de ondas eletromagnéticas.

Podem existir várias portadoras de rádio em um mesmo meio, sem que haja interferência entre elas. Para que os dados sejam recuperados, o receptor sintoniza em uma frequência determinada e descarta as outras portadoras de diferentes frequências.

Geralmente, o transceptor, dispositivo transmissor/receptor, ou *access point* é conectado a uma rede Ethernet. Além do *access point* fornecer a comunicação com a rede Ethernet, ele também intermedia o tráfego com outros *access points*, por meio do sistema de *roaming*.

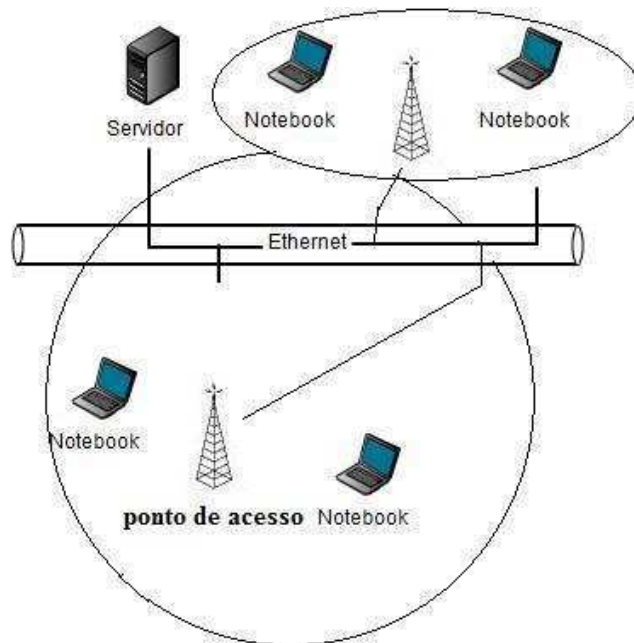


Figura 2.9: Redes WLAN

2.3.3. WMAN

As WMANs – *Wireless Metropolitan Area Network* – são de uso corporativo e atravessam cidades e estados.

A WMAN faz a interconexão de redes WLAN com a Internet. Além disso, as WMANs podem funcionar como backups das redes cabeadas.

Entre os padrões de WLANs o mais importante é o WiMAX, que tem alta velocidade e controle distribuído para múltiplos serviços.

O WiMAX possui técnicas implícitas de segurança e de criptografia por meio de um processo de autenticação de uma estação que entra na rede de um processo em que ocorre o gerenciamento de

chaves de criptografia e autenticação de dados, essas técnicas estão no NS2 – Network Simulation.

2.3.4. WWAN

As WWANs – *Wireless Wide Area Network* – são redes que estabelecem comunicações maiores que as WMANs, abrangendo países, através do uso de sistema de satélites.

Escolas e comércios que se situam em áreas rurais são grandes beneficiários das WWANs, já que podem ficar bem afastados das grandes cidades e não estarem ao alcance das WLANs ou das WMANs.

Alguns exemplos de WWANs são celulares digitais, modems via satélite, computadores ligados com cartões WWAN.

2.3.5 WLL

Para se falar um pouco sobre essa tecnologia, baseou-se no artigo de José Maurício Santos Pinheiro [19]. Dessa forma, podemos dizer que o WLL – *Wireless Local Loop* – chega como alternativa aos problemas que as empresas que prestam serviços de telecomunicações enfrentam com relação a implantação de redes de telefonia fixa com cabeamento tradicional.

O WLL é uma tecnologia de telecomunicações que usa radiofrequência como forma de estabelecer a ligação de um telefone à central telefônica pública, não sendo necessário assim o uso de cabeamento para a comunicação.

O funcionamento da rede se dá pelo seguinte: ocorre a ligação do telefone do usuário ao equipamento de rádio, este troca informações com uma estação de rádio base, que converte os sinais de rádio em sinais que a central telefônica possa entender, então a chamada se estabelece normalmente.

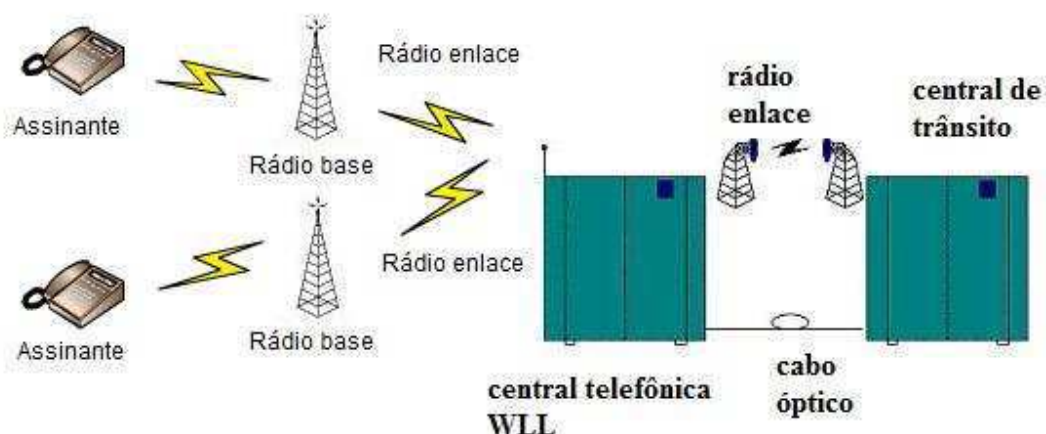


Figura 2.10: Acesso WLL

A central concentradora, que fica localizada próxima ou dentro da central de telefonia, é a responsável pela conversão dos sinais recebidos pela estação rádio base em informações para a central telefônica. Essa central pode também estar ligada à central pública por meio de microondas ou fibras ópticas.

Cada assinante do sistema possui um transceptor que permite a conexão do telefone ao concentrador, estabelecendo assim a conexão wireless.

A rede convencional de telefonia oferece um ótimo serviço de comunicação, com qualidade de voz e bom tráfego de dados, mas graças aos vários estágios da rede cabeada, podem ocorrer

problemas na transmissão, atenuações e dificultar a utilização de outros serviços, como internet banda larga, por exemplo.

O WLL apresenta instalação rápida e eficiente, além de ter baixo custo de manutenção e de implantação quando comparado às redes convencionais. A capacidade de expansão também apresenta grande diferença entre os dois tipos de redes, O WLL pode se expandir muito rapidamente, ao contrário da rede tradicional.

3. NETWORK SIMULATOR

NS ou Network Simulator é um simulador de redes de computadores popular nos meios acadêmicos por ter o código fonte aberto. A primeira versão do NS que surgiu em 1989 era uma variação do software REAL Network Simulator, um projeto da *Cornell University*, EUA. A partir de 1995 o NS passou a ter o apoio do DARPA (*Defense Advanced Research Projects Agency*) [10] através do projeto SAMAN (*Simulation Augmented by Measurement and Analysis for Networks*) [11] e pela NSF (*National Science Foundation*, EUA) através do projeto CONSER [12], em colaboração com outros pesquisadores como o centro ICIR (*The International Computer Science Institute Center for Internet Research*) [13]. O NS também sempre inclui contribuições substanciais de outros pesquisadores, incluindo códigos de redes sem fio do UCB Daedelus e dos projetos do CMU Monarch e da Sun Microsystems.

A programação do NS é feita em nas linguagens: C++ e OTCL (*Object-oriented Tool Command Language*). O motivo da utilização de duas linguagens diferentes baseia-se no fato de que de um lado existe a necessidade de uma linguagem mais robusta para a manipulação de bytes, pacotes e para implementar algoritmos que rodem um grande conjunto de dados e nesse contexto C++, que

é uma linguagem compilada e de uso tradicional, mostrou-se mais eficiente. De outro lado é fato que, durante o processo de simulação, ajustes são necessários com certa frequência. Muda-se o tamanho do enlace e faz-se um teste, muda-se o atraso e faz-se um teste, acrescenta-se um nó e faz-se um teste. Enfim, haveria um desgaste muito grande se, a cada mudança de parâmetro, e elas são muitas em uma simulação, houvesse a necessidade de se compilar o programa para testá-lo [7]. A utilização da linguagem interpretada evita o desgaste por parte do usuário, pois há uma simplificação no processo interativo de ajustes na estrutura da simulação.

3.1.FUNCIONAMENTO DO NETWORK SIMULATOR

Para executar as simulações no NS-2 é necessário definir os elementos da rede, os protocolos utilizados e finalmente inserir o tráfego. Essas definições são implementadas em scripts escritos na linguagem OTCL. O básico para a definição de uma rede e sua simulação é a definição dos nós, enlaces, cenário e tráfego. Não é objetivo deste trabalho detalhar a linguagem OTCL, portanto, será apresentado somente o básico para o entendimento da aplicação.

3.1.1. Script TCL

No apêndice “A” encontra-se um exemplo de um script, gerado na aplicação proposta nesse trabalho, escrito na linguagem OTCL para simulação de uma rede LAN utilizando o Network Simulator. Os arquivos devem ser gravados com a extensão “.tcl”.

3.1.2. Declaração de Nós

A estrutura de um nó no NS é definida pelo comando:

```
set <nome do Nó> [$ns node]
```

É possível vincular mais de um agente ou protocolo a um mesmo nó para tornar a simulação mais interessante.

3.1.3. Declaração de Enlaces

Um enlace ou link é uma estrutura física que conecta os nós formando a topologia da rede. Os enlaces são definidos pelo comando

```
$ns duplex-link $Brasilia $Recife 1Mb 5ms DropTail
```

onde *duplex-link* indica que o enlace é do tipo duplex, Brasília é o nó de origem, Recife é o nó de destino, 1Mb é a velocidade do nós em Megabits por segundo, 5ms é o atraso do link em milissegundos e *DropTail* é a política de fila.

3.1.4. Cenário e Tráfego

Para inserir o tráfego na simulação é necessário definir o agente ou protocolo de transporte e o tipo de aplicação que fará uso desse transporte. Como exemplo de protocolo de transporte vamos mostrar a utilização do UDP (*User Datagram Protocol*) e como exemplo do tipo de aplicação vamos utilizar o tipo CBR (*Constant Bit Rate*).

Criando um agente UDP em um nó:

```
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0
```

Definindo o tráfego CBR associado ao agente UDP:

```
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

3.2.NAM - NETWORK ANIMATOR

Como a interface de criação e manipulação dos dados das simulações é baseada em arquivos de texto, a ferramenta para visualização gráfica das simulações é o NAM (*Network Animator*). O NAM é uma ferramenta de animação que interpreta os arquivos de extensão .nam gerados pelo NS2 de uma simulação executada e mostra em um ambiente gráfico a topologia, transmissão dos fluxos, a formação de filas, o descarte de pacotes, enlaces tornando-se não operacionais, enfim, todo o comportamento dos fluxos de aplicações.

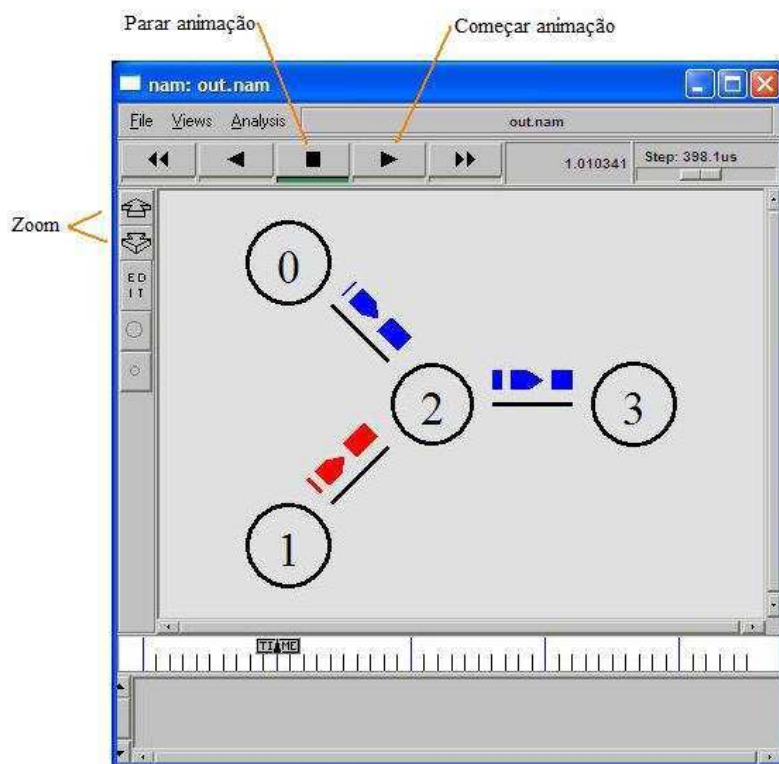


Figura 3.1: Tela do NAM com uma animação em andamento

Através dos comandos do console da interface da ferramenta o usuário controla a velocidade, trecho e sentido da animação. Segurando e arrastando o retângulo “*TIME*” é possível colocar a animação em qualquer instante de tempo.

4. NETWORK SIMULATOR EM AMBIENTE WEB

A idéia de fornecer um ambiente onde seja possível utilizar o NS-2 remotamente é interessante pelo fato da complexidade de instalação do Simulator, o não funcionamento do simulador em sistema operacional Windows e pela possibilidade de utilização por empresas ou universidades que não tenham o simulador instalado em suas máquinas e necessitem validar um arquivo tcl. Esse projeto é totalmente original uma vez que em nossas pesquisas não foi encontrado nenhum trabalho semelhante para utilização do Network Simulator remotamente.

A solução definida no projeto para possibilitar a simulação de redes em ambiente WEB foi a utilização de um servidor Apache 2 e PHP 5 rodando em sistema operacional Ubuntu versão 8.04 com o Network Simulator versão 2.27 e hospedando paginas HTML e PHP. Além disso é necessário a cópia dos arquivos executáveis nam.exe e wget.exe em uma pasta criada no diretório raiz c:\.

Ubuntu é uma distribuição do sistema operacional Linux baseado na distribuição Debian. Sua distribuição, livre e gratuita, é patrocinada pela Canonical Ltd sendo uma das mais populares dentro dos utilizadores de Linux.

Conforme citado anteriormente, NS ou Network Simulator é um simulador de redes de computadores popular nos meios acadêmicos por ter o código fonte aberto. Suporta os protocolos de rede mais populares tanto para redes cabeadas quanto para redes sem fio.

HTML ou Linguagem de Marcação de Hipertexto, é uma linguagem de marcação utilizada para produzir páginas na Web. Basicamente trata-se de um conjunto de etiquetas (tags) que servem para definir a forma na qual se apresentará o texto e outros elementos da página. O browser ou navegador interpreta documentos HTML para mostrar as informações em um ambiente gráfico. Para produzir páginas dinâmicas foi utilizado PHP, linguagem de programação interpretada e livre. O PHP possibilita escrever scripts inseridos no HTML sendo que tudo é processado no servidor antes de ser enviado ao cliente. O processamento dos códigos em PHP ficam por conta do servidor apache 2. A arquitetura está apresentada na figura 4.1 abaixo.

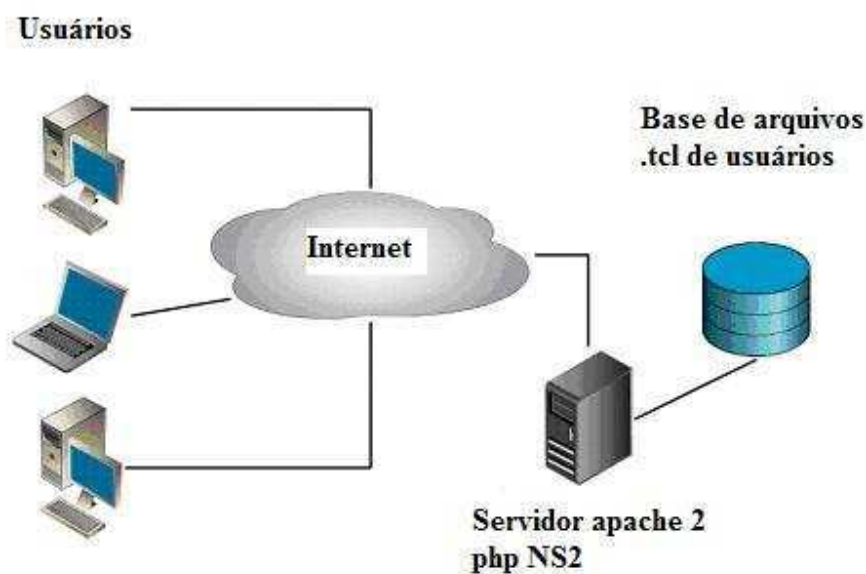


Figura 4.1 Arquitetura do projeto

O nam.exe é um executável do software NAM que pode ser encontrado no site oficial do Network Animator desenvolvido para rodar em plataforma Microsoft Windows. A versão utilizada no projeto é a Nam 1.0a11a de tamanho 1.8 Mb.

A troca de arquivos de extensão nam gerados no servidor da aplicação para o cliente é feita pelo programa livre wget.exe que propicia *download* de dados da web. O Wget faz parte do projeto GNU e já foi portado para ambientes Mac OS X e Microsoft Windows. Seu *download* pode ser feito no site oficial do projeto GNU.

4.1.FUNCIONAMENTO DO SIMULADOR EM AMBIENTE WEB

Após a inicialização do servidor, o usuário conecta-se via browser ao endereço do servidor. A figura mostra a página inicial da aplicação onde temos as opções de simulação de uma rede LAN, simulação de uma rede WLAN ou então o usuário pode criar seu próprio arquivo de extensão tcl para simulação.

4.1.1. Início

A figura mostra a página inicial da aplicação onde no menu à esquerda temos as opções de “Início”, “Simulação de LAN”, “Simulação de WLAN”, “Upload de Arquivo tcl”, “Projeto” e “Links”. A opção “Início” sempre remete o usuário à tela inicial da aplicação conforme vemos na figura 4.2.

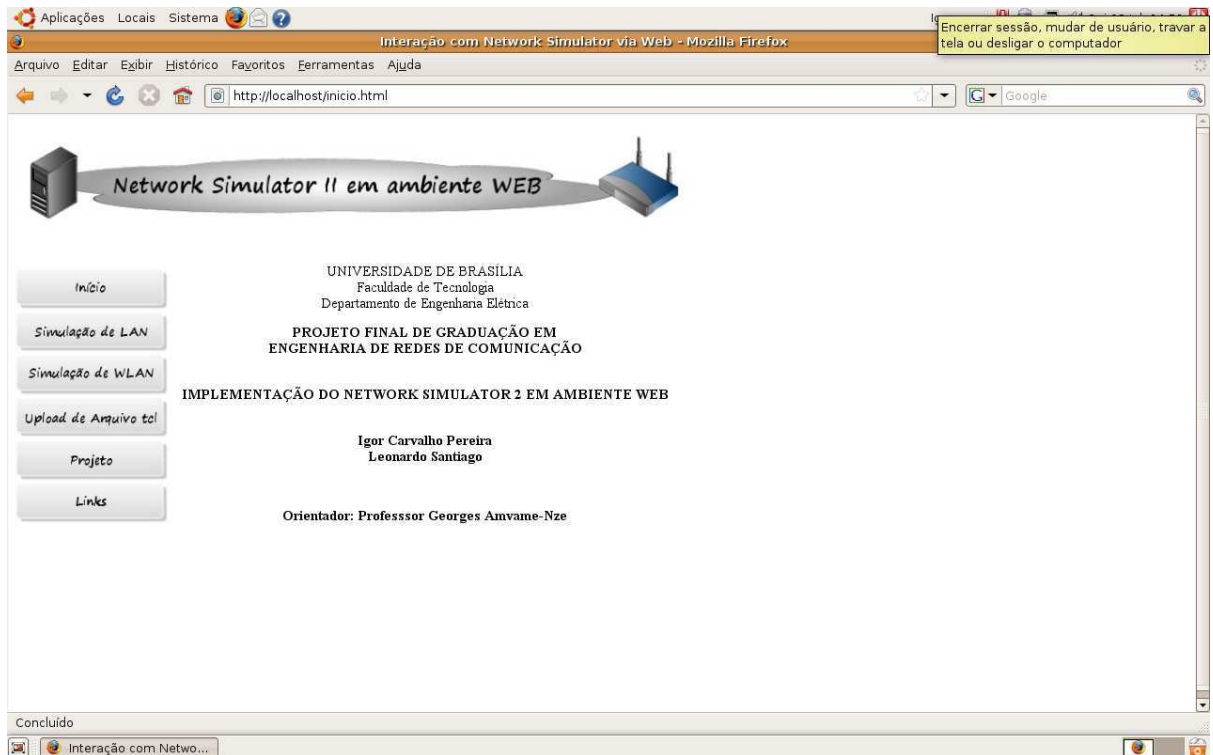


Figura 4.2: Tela inicial da aplicação

4.1.2. Simulação de LAN

A opção “Simulação de LAN” permite que o usuário possa interagir com o Network Simulator sem a necessidade de entender a linguagem OTCL. Ao escolher as opções pré-definidas de nós e preencher parâmetros para enlances e tráfego, o software gera de maneira automática o arquivo de extensão tcl que será interpretado pelo simulador. Ao selecionar a opção de simulação de LAN, o usuário encontrará a tela mostrada na figura 4.3.

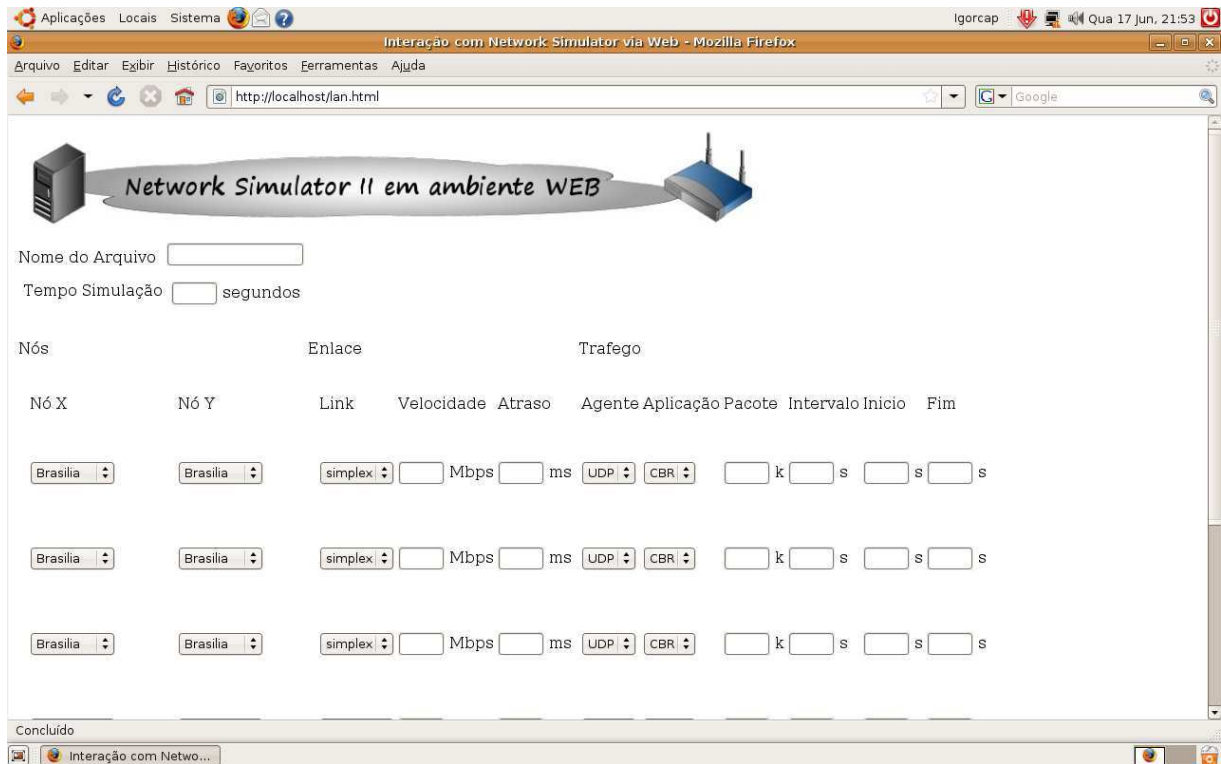


Figura 4.3: Simulação de LAN

Por trás da página lan.html está um formulário HTML onde todos os campos preenchidos são entregues ao módulo submete.php para tratamento dos dados. O primeiro passo é preencher no campo nome do arquivo para o usuário identificar seu documento posteriormente no servidor. O tempo de simulação é a quantidade de segundos que o NS-2 vai gerar dados para que o NAM possa mostrar a animação. É possível a configuração de uma LAN com até cinco nós e quatro enlaces de dados. As opções pré-definidas de nós são Brasília, Curitiba, Recife, Rio de Janeiro e São Paulo. Ao escolher os nós o sistema não permite que enlaces comecem e terminem no mesmo ponto e nem que sejam selecionados enlaces repetidos. Caso alguma dessas situações ocorra, ao clicar em cadastrar será gerada uma mensagem de erro explicativa ao usuário. Pode-se escolher links simplex ou duplex. A velocidade e o atraso dos links são informados nas unidades

megabits por segundo (Mbps) e milissegundos (ms) respectivamente. Na figura 4.4 vemos a tela preenchida.

Nós		Enlace			Tráfego					
Nó X	Nó Y	Link	Velocidade	Atraso	Agente	Aplicação	Pacote	Intervalo	Inicio	Fim
Brasília	Curitiba	simplex	1 Mbps	1 ms	UDP	CBR	500 k	0,005 s	0 s	4 s
Brasília	Recife	simplex	1 Mbps	1 ms	UDP	CBR	500 k	0,005 s	0,5 s	5 s
Brasília	Rio	simplex	1 Mbps	1 ms	UDP	CBR	500 k	0,005 s	0,5 s	5 s
Rio	Saopaulo	duplex	10 Mbps	2 ms	UDP	CBR	500 k	0,01 s	1 s	4,5 s

Figura 4.4: Exemplo de Simulação de LAN com parâmetros preenchidos

Para o tráfego existem as opções de agente UDP e TCP e para aplicação as possibilidades são CBR e FTP. Ao finalizar a configuração da rede LAN para simulação deve-se apertar o botão cadastrar que acionará o módulo submete.php.

A primeira providência do script acionado é carregar os dados capturados no formulário HTML pelo método POST. A seguir é feita uma série de validações dos dados recebidos como, por exemplo: Se os campos obrigatórios vieram preenchidos, se no servidor já existe algum arquivo

com o nome utilizado, se os enlaces tem início e fim em nós diferentes e se não estão está ocorrendo repetição de enlaces.

Para criar o arquivo tcl o módulo submete.php cria um novo arquivo através do comando

```
$cria = fopen($arquivo . ".tcl", "w+");
```

e em seguida, vai concatenando o conteúdo do arquivo na variável *\$dados*:

```
$dados .= "#Create a simulator object".PULA;  
$dados .= "set ns [new Simulator]".PULA;  
$dados .= " ".PULA;
```

A utilização da string “.PULA” no final de cada linha é para quebra de linha dentro do arquivo tcl. A string “PULA” é uma constante definida no começo do script pelos caracteres (13) + (10) da tabela ASCII que representam a quebra de linha. A terceira linha do código acima é para deixar uma linha em branco no arquivo tcl para uma melhor compreensão do programa OTCL.

Para gravar o conteúdo concatenado na variável *\$dados* no arquivo tcl e fechá-lo é utilizado o código abaixo:

```
$escreve = fwrite($cria,$dados);  
fclose($cria);
```

Se houver alguma falha no preenchimento, o sistema mostrará uma mensagem de erro sinalizando onde está o problema. Caso contrário, será apresentada a tela da figura abaixo.

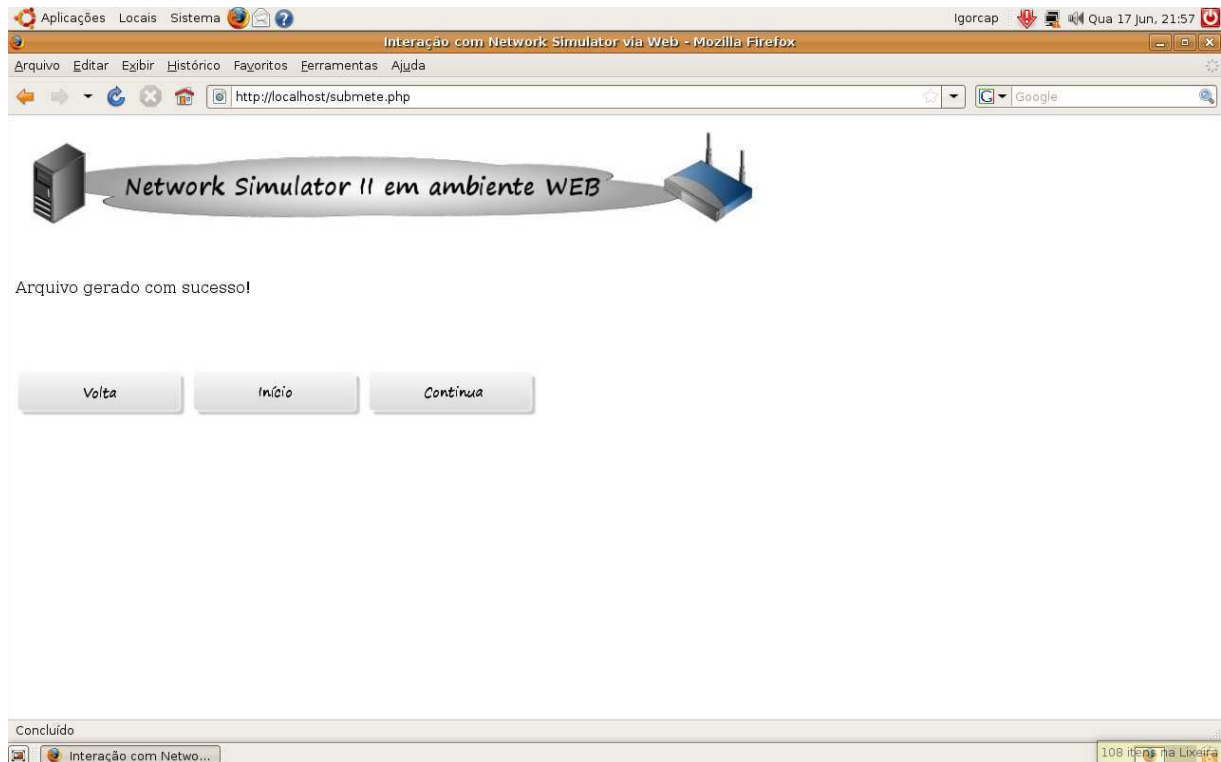


Figura 4.5: O arquivo de extensão tcl está pronto para ser submetido ao NS

Se o usuário deseja fazer alguma modificação na configuração de sua LAN pode clicar em voltar para alterar algum parâmetro. Se tudo ocorreu corretamente, ao clicar em continua o sistema exibirá o arquivo tcl que foi gerado possibilitando que o usuário tenha acesso à estrutura da linguagem OTCL. O módulo arquivo.php é o responsável por mostrar no browser o conteúdo do arquivo tcl e o código PHP contendo o *loop* que imprime os dados na tela encontra-se abaixo.

```
for($i=0;$i<$tot;$i++) {  
    $ct=$txt[$i][0];  
    $linha = Chop($txt[$i]);  
    print strip_tags($linha);  
    print "<br>";  
}
```

}

A figura 4.6 mostra a visão do usuário do conteúdo do arquivo tcl que será submetido ao NS-2.

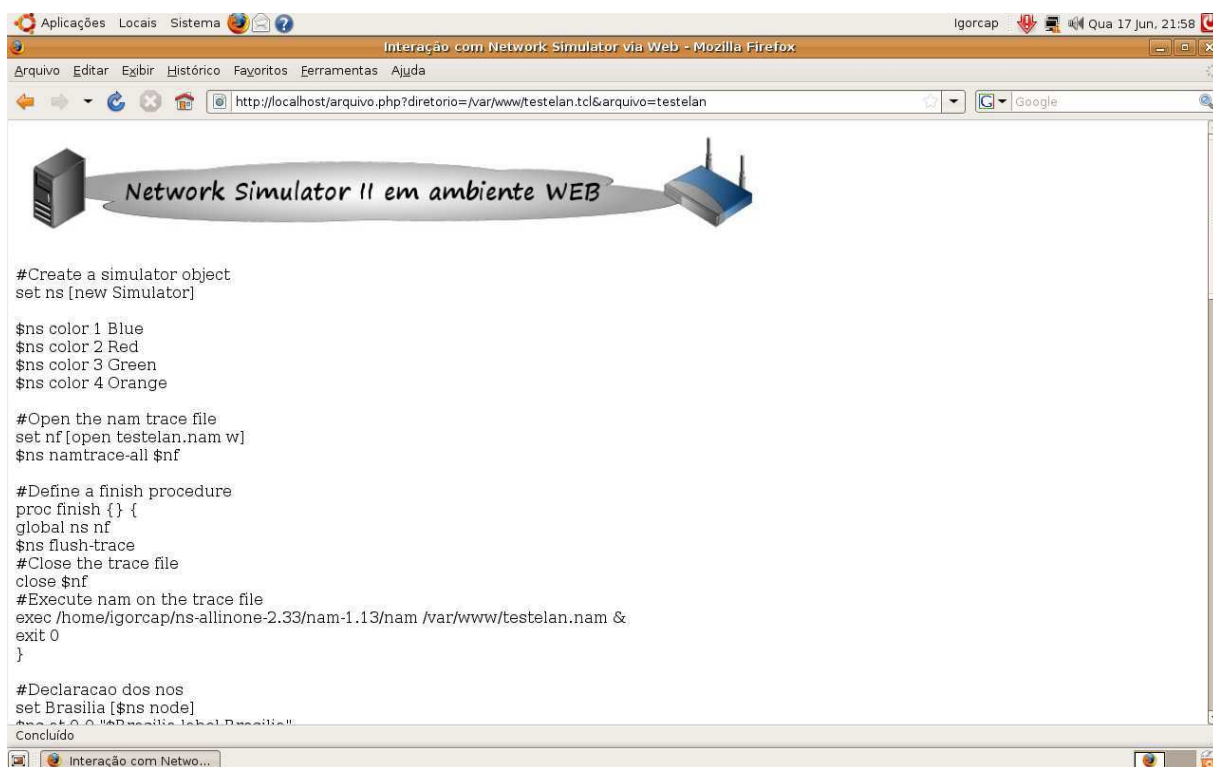


Figura 4.6: Visualização do arquivo tcl gerado pela aplicação

Após a visualização do arquivo gerado de forma automática pelo software, o usuário ainda tem a opção de voltar para eventuais modificações na configuração de sua LAN ou então deve apertar o botão “Continua” para submeter o arquivo tcl ao Network Simulator 2 instalado no servidor da aplicação. A próxima tela exibida contém um link para olhar ou salvar o arquivo de extensão nam gerado pelo NS e também um link para visualizar a animação da rede LAN no NAM.

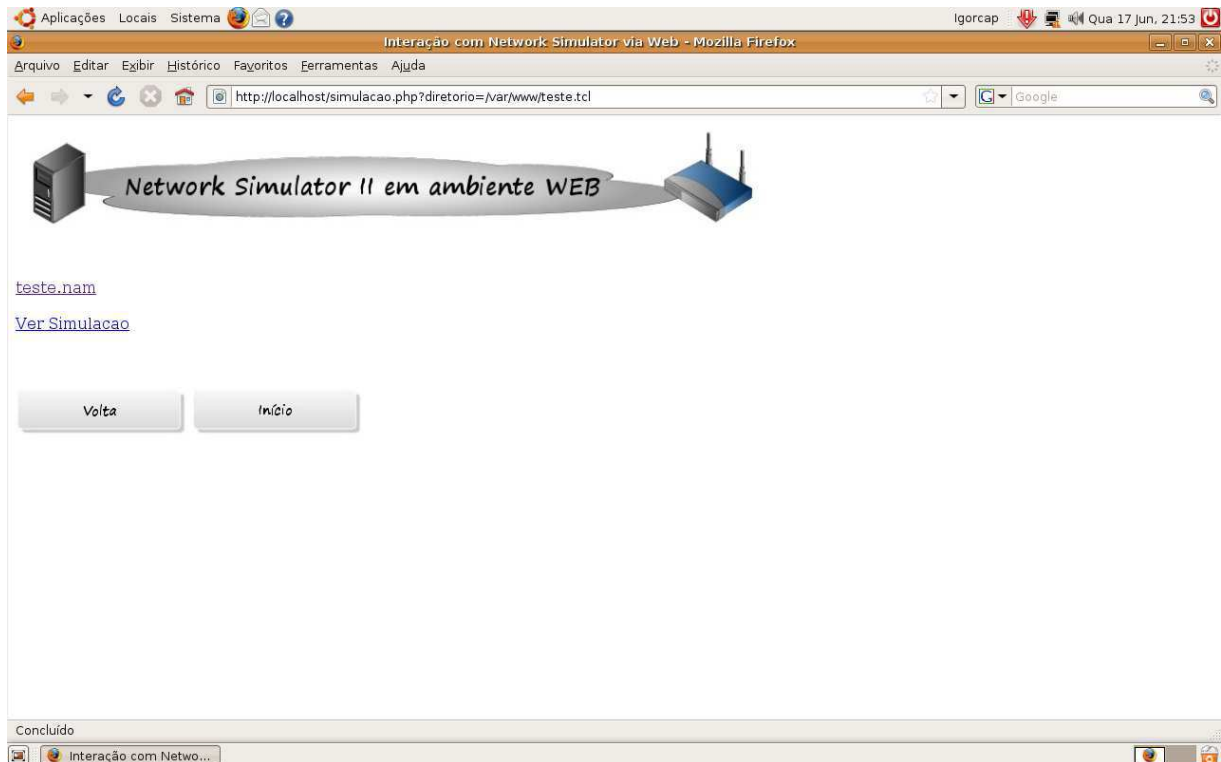


Figura 4.7: Opções de visualizar o conteúdo do arquivo .nam ou assistir a animação no NAM.

O servidor sempre gera dois arquivos iguais de extensão nam. Um deles com o mesmo nome do arquivo tcl que ficará disponível para download conforme mostrado na figura 4.7 e outro sempre com o nome out.nam que será utilizado para a execução da animação no cliente.

Escolhendo a opção “Ver Simulação” irá surgir uma janela conforme a figura 4.8 que pede permissão ao usuário para executar o script ns.bat em sua máquina.

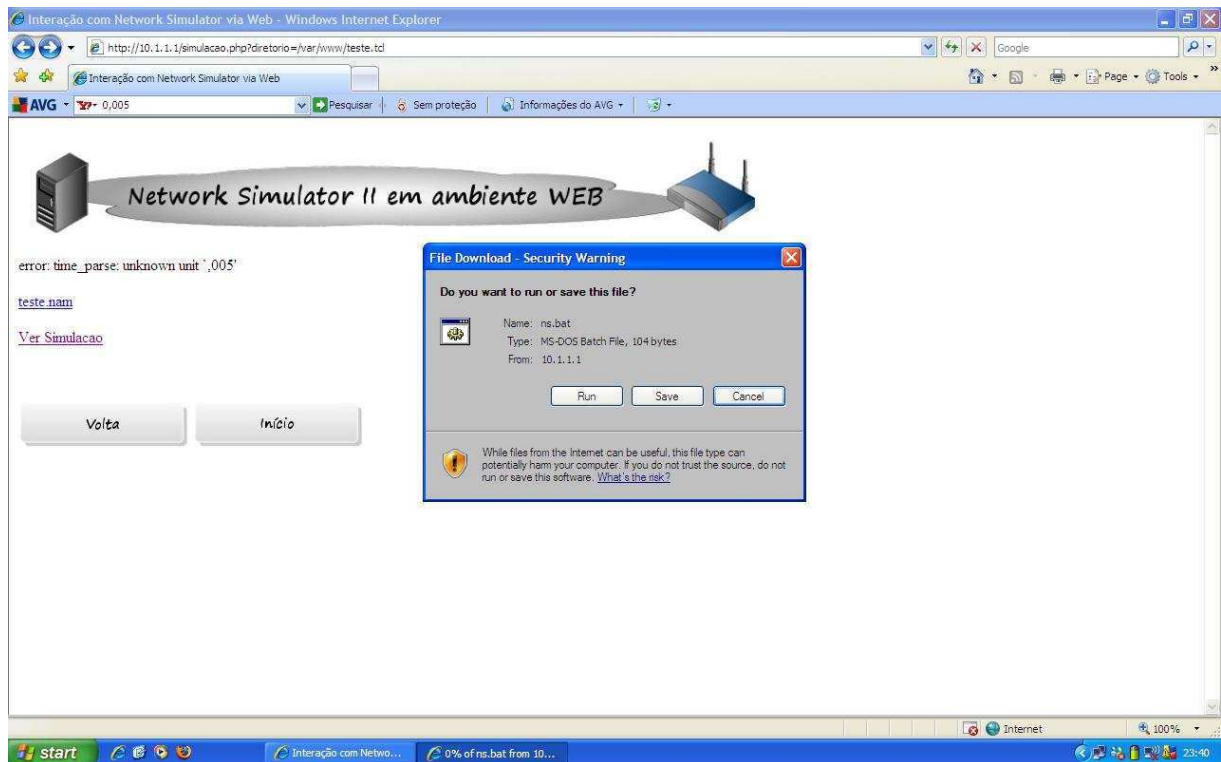


Figura 4.8: Aceitar a execução da simulação

Apertando a opção “run” o script ns.bat executará no computador do cliente os comandos:

```
cd C:\tmp\  
del out.nam  
wget http://10.1.1.1/out.nam  
nam out.nam
```

A segunda linha do código acima apaga, caso exista, algum outro arquivo out.nam copiado em alguma simulação anterior, a terceira linha utiliza o wget para copiar o novo arquivo out.nam do servidor para o cliente e a quarta linha executa o nam tendo como entrada o novo arquivo out.nam.

Após a execução do arquivo ns.bat a simulação iniciará automaticamente no computador cliente e a tela do NAM será exibida conforme mostra a figura 4.9.

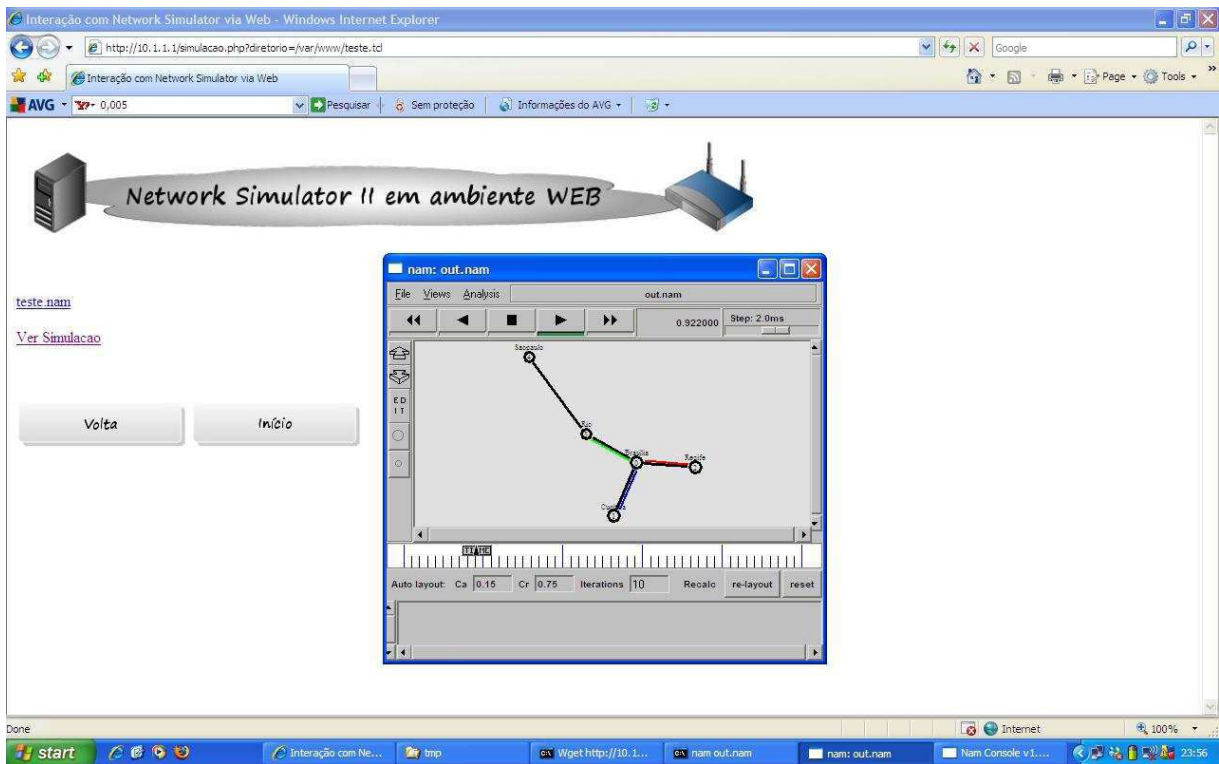


Figura 4.9: NAM executando a animação

Os controles do NAM devem ser utilizados para controlar a apresentação da animação.

4.1.3. Simulação de WLAN

O link “Simulação de WLAN” leva o usuário a simulação de redes sem fio pré-definida no Network Simulador 2 em ambiente WEB. A figura abaixo mostra a tela observada pelo usuário ao escolher esta opção.

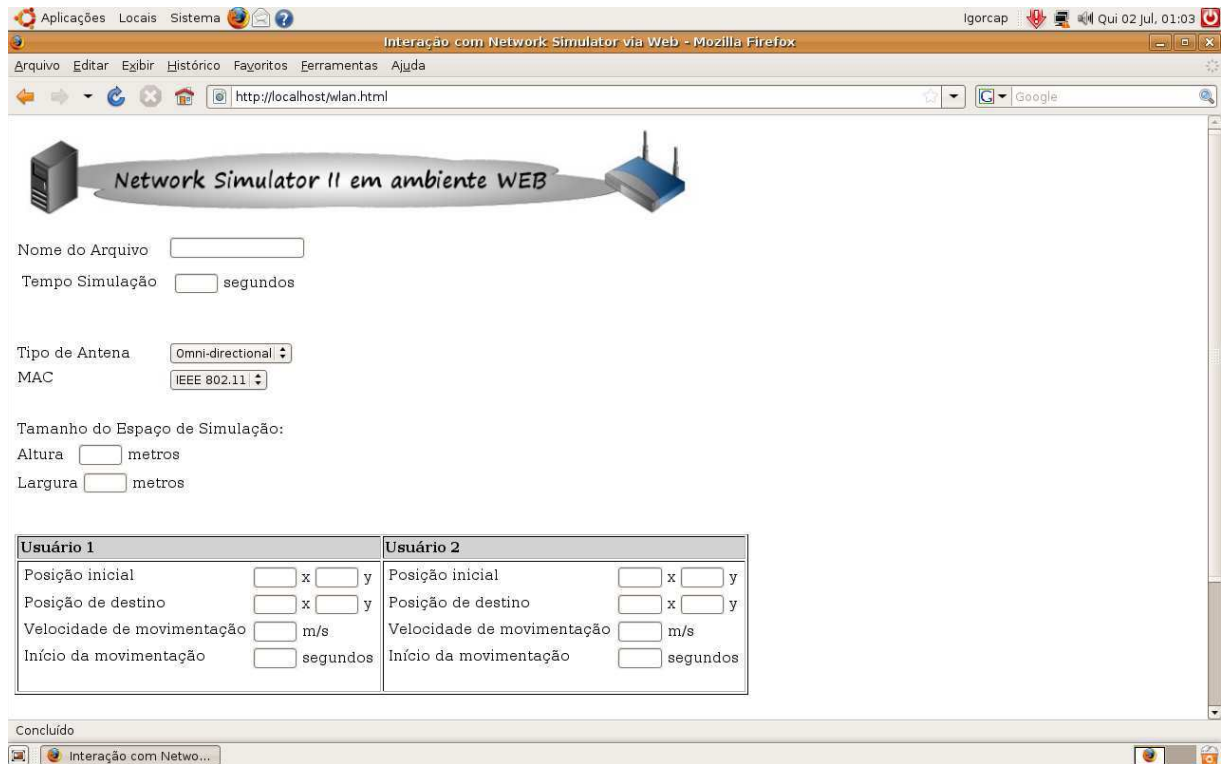


Figura 4.10: Simulação de WLAN

As possibilidades de interação são limitadas a uma rede com dois nós constituída pelo usuário 1 e o usuário 2. O tipo de antena utilizado é a omnidirecional e o protocolo utilizado é o MAC do padrão IEEE 802.11. O site permite que sejam alterados parâmetros do tamanho da área utilizada na simulação da rede sem-fio e da movimentação espacial dos nós. Considerando essas especificações, o arquivo tcl é construído empregando as configurações definidas abaixo:

```
# =====
# Opções Definidas
# =====
set val(chan) Channel/WirelessChannel; # Channel type
```

```

set val(prop) Propagation/TwoRayGround;      # Radio-propagation
set val(ant) Antenna/OmniAntenna;           # Antenna type
set val(ll) LL;                              # Link layer type
set val(ifq) Queue/DropTail/PriQueue;       # Queue type
set val(ifq) 50;                             # Max packet in ifq
set val(net) Phy/WirelessPhy;               # Net interface type
set val(mac) Mac/802_11;                    # MAC type
set val(rp) DSDV;                            # ad-hoc protocol
set val(nn) 2;                               # number of nodes

```

A motivação da implementação desta área destinada a WLAN é mostrar que o Network Simulator 2 também pode ser utilizado para simulação de redes sem-fio. Esse primeiro contato desperta no usuário a curiosidade de pesquisar e assim construir seus próprios arquivos tcl mais elaborados para simulação de redes sem fio e submetê-los ao NS-2 através da opção upload de arquivo que será abordada na próxima seção.

4.1.4. Upload de arquivo tcl

A aplicação permite que o usuário possa criar seu próprio arquivo tcl para submeter ao NS instalado no servidor via ferramenta WEB. Essa funcionalidade é de extrema utilidade para usuários que já conhecem a estrutura de arquivos escritos na linguagem OTCL e desejem utilizar remotamente o Network Simulator 2 para validar seus arquivos e gerar um arquivo de extensão nam para ver a animação de sua simulação. Para utilizar essa opção aperta-se no botão upload de

arquivo tcl no menu inicial. A tela exibida na figura 4.10 mostra o que será exibido para o usuário na opção upload de arquivo tcl.

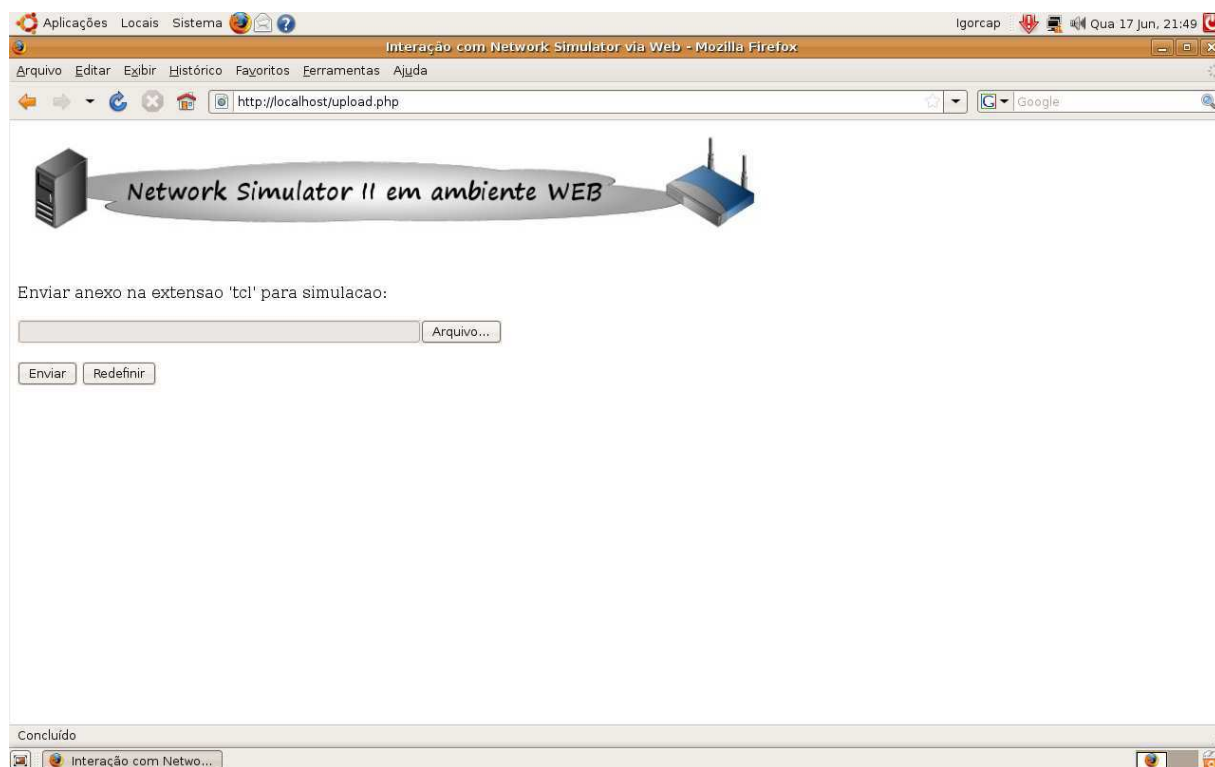


Figura 4.11: Opção de upload de arquivo tcl

Deve-se apertar no botão “Arquivo...” para selecionar a localização do mesmo no computador cliente. Existe uma validação efetuada pelo sistema onde somente serem aceitos arquivos de extensão tcl. Caso a localização do arquivo selecionado esteja errada, existe a opção redefinir onde é possível escolher um novo arquivo. Após selecionar o arquivo preterido clica-se em “Enviar” para que o arquivo de extensão tcl seja encaminhado para o servidor. A figura 4.11 mostra um arquivo já selecionado para upload.

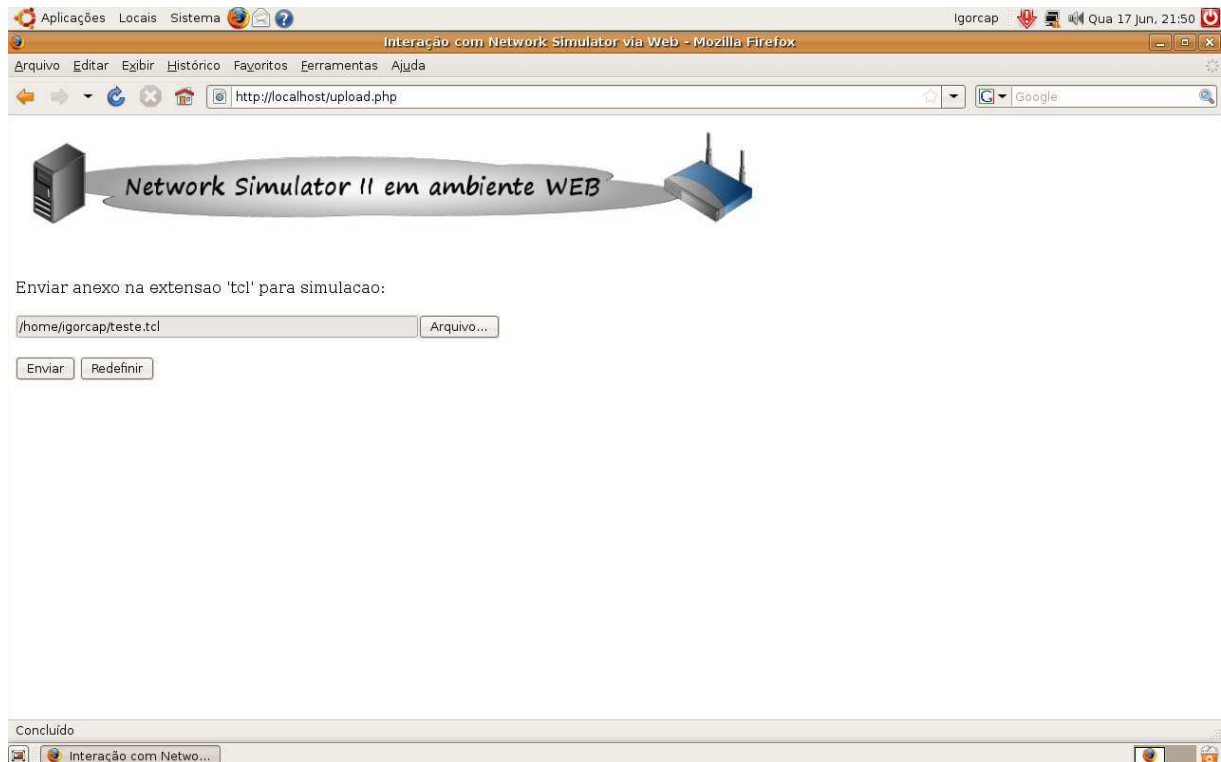


Figura 4.12: Enviando arquivo para upload

O código fonte abaixo retirado do componente upload.php, mostra os comandos utilizados em PHP para indicação da pasta onde está o arquivo será salvo no servidor, validação da extensão e upload do mesmo.

```
$caminho = "/var/www/";  
$caminho = $caminho.$_FILES["DfAnexo"]["name"];  
if(ereggi(".tcl$", $_FILES["DfAnexo"]["name"])) {  
    copy($_FILES["DfAnexo"]["tmp_name"], $caminho);  
}  
else {  
    echo "Erro no envio do anexo<br /><br />";  
}
```

```
echo "Extensao invalida de arquivo!";  
    exit;  
}
```

A aplicação é hospedada no Linux na pasta `/var/www/`. É neste local que o servidor Apache disponibiliza as páginas que serão visualizadas em outros computadores via browser. O arquivo deve ser gravado no servidor com a permissão de leitura para o usuário `www-data` que é quem irá acessar o arquivo posteriormente execução do Network Animator. No Linux, aplicações WEB são executadas por esse usuário.

A função `eregi` no PHP compara se a string `".tcl"` é encontrada no nome do arquivo selecionado para *upload*. Essa validação da extensão do arquivo independe se está escrito `.tcl` ou `.TCL` pois a função é *case insensitive*, ou seja, não distingue caracteres maiúsculo de minúsculos.

Se o *upload* ocorreu corretamente será apresentada a tela da figura 4.12.

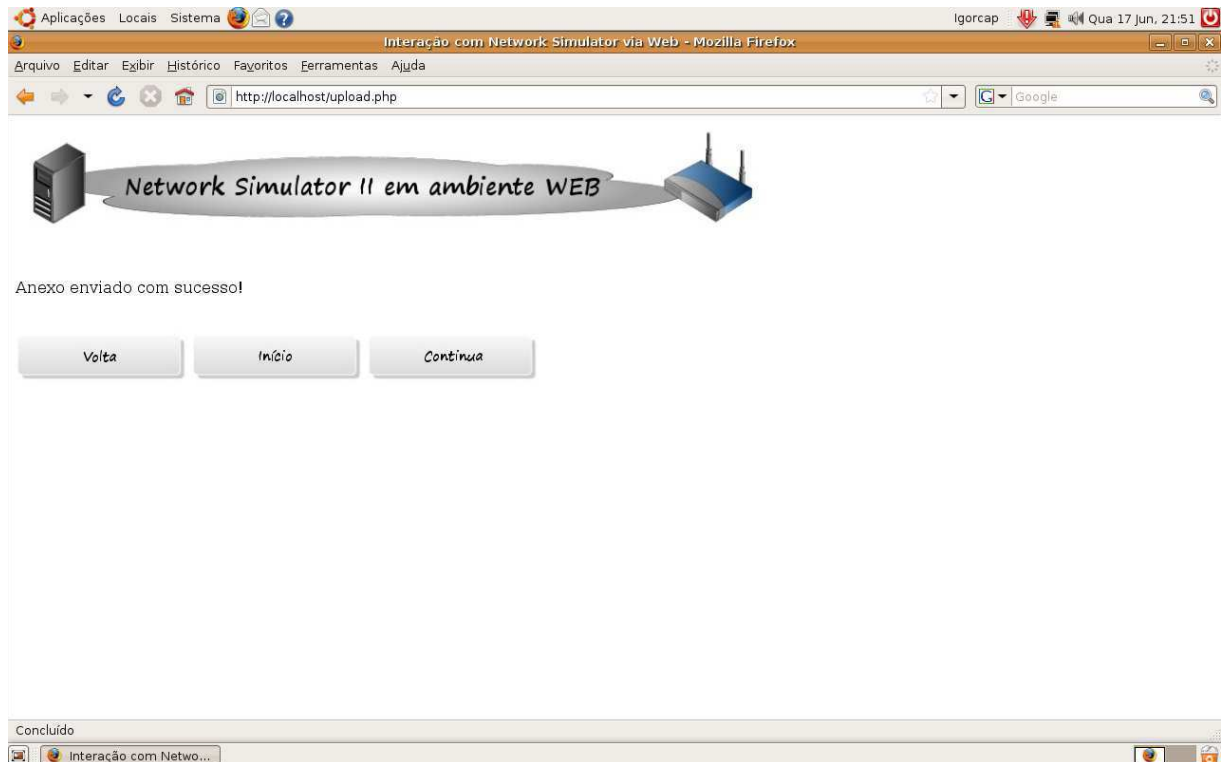


Figura 4.13: Arquivo enviado com sucesso para o servidor.

Clicando em “Continua” o usuário terá acesso ao conteúdo do arquivo tcl que foi submetido a *upload* conforme a tela já mostrada anteriormente na figura 4.6.

4.1.5. Projeto

No menu inicial temos a opção projeto onde será encontrado maiores informações sobre o projeto desenvolvido neste trabalho e algumas instruções para os usuário do Network Simulator II em ambiente WEB. A figura 4.13 ilustra o que será observado ao escolher a opção projeto.

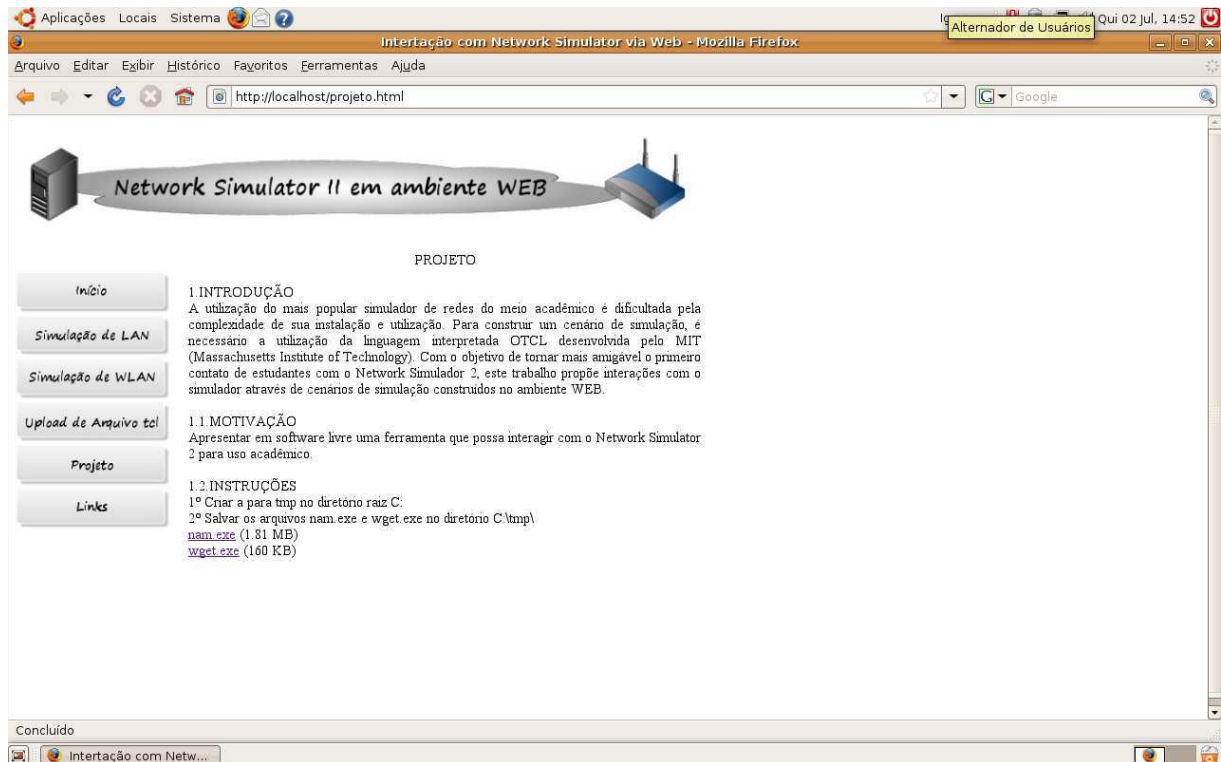


Figura 4.14: Detalhes sobre o projeto e instruções

Também está disponível nessa área os arquivos `nam.exe` e `wget.exe` necessários para executar a animação do NAM no computador cliente.

4.1.6. Links

Selecionando a opção Links do menu inicial, o usuário terá acesso a alguns links que julgamos relevantes ao projeto. Conforme a figura, os links encontrados são os endereços do site oficial do NS, site oficial do NAM, projeto do NS-3, Universidade de Brasília e Departamento de Engenharia Elétrica da Universidade de Brasília.

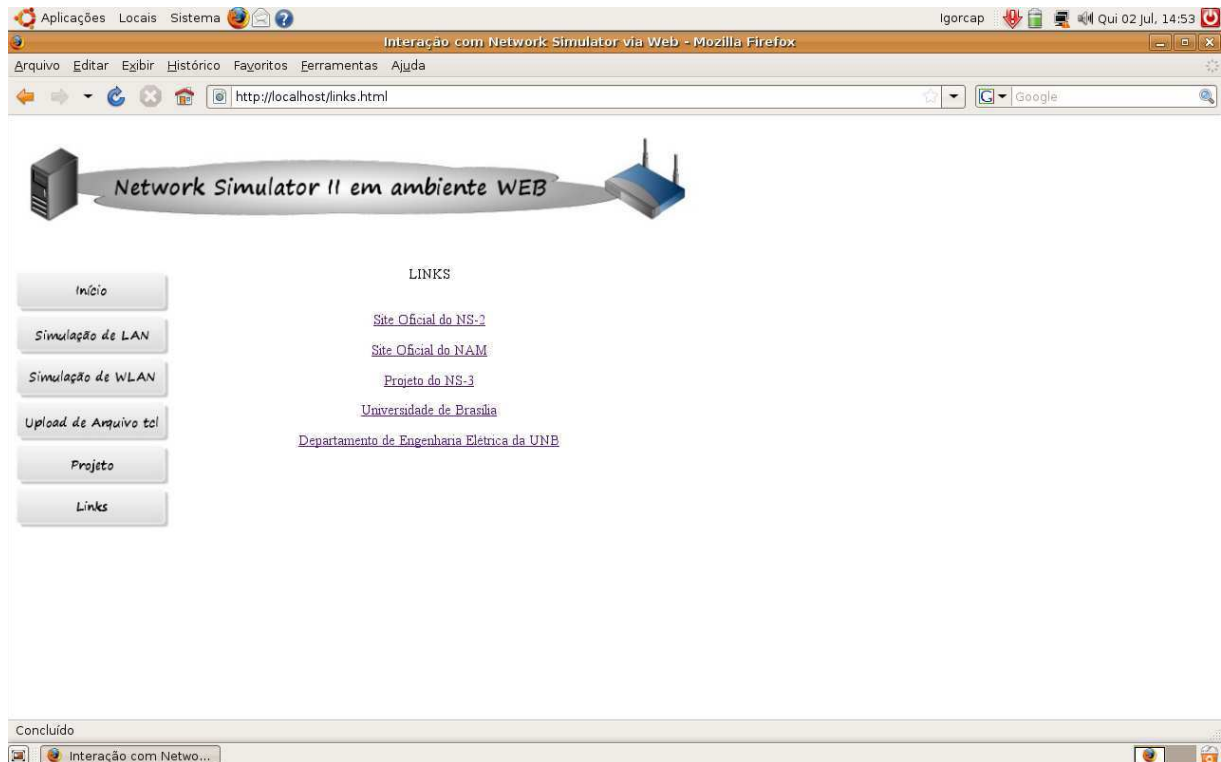


Figura 4.15: Links relevantes ao projeto

5. CONCLUSÃO

A utilização de simuladores de redes é imprescindível para testar e validar modelos propostos antes de sua efetiva implantação. O Network Simulator 2 na condição de software livre é mantido e atualizado por diversos colaboradores que estão sempre possibilitando que novos protocolos e novas tecnologias possam ser simuladas. Esse suporte de maneira rápida e colaborativa é um dos segredos do sucesso do NS que tem o posto de um dos mais utilizados simuladores de rede. A implementação do Network Simulator 2 em ambiente WEB proposta nesse trabalho é mais uma colaboração para facilitar e popularizar o uso do software no meio acadêmico.

O projeto do Network Simulator 2 em ambiente WEB permite que o primeiro contato de estudante com o NS-2 seja mais sutil e menos traumático porque além de o usuário não precisar instalar o simulador em um ambiente Linux, também é possível gerar arquivos tcl através de seleções de campos e preenchimento de valores em caixas de texto. Outra grande inovação é a possibilidade de usuários já familiarizados com a linguagem OTCL criarem seus arquivos em qualquer editor de texto e gerarem através do site seus arquivos nam e visualizarem suas simulações de forma remota e locais onde não exista o ambiente pronto para utilização do Network Simulator 2.

5.1. TRABALHOS FUTUROS

Sugerimos que os trabalhos futuros que procurem dar continuação a este trabalho incluam um ou mais dos seguintes quesitos:

- a) Aperfeiçoar as opções pré-definidas de simulação de LAN e WLAN permitindo um maior número de possibilidades de simulação de redes para o usuário;
- b) Fazer um tratamento minucioso para depurar erros de programação em arquivo tcl enviados ao servidor através da opção upload de arquivo tcl;
- c) Desenvolver através de stream de vídeo uma maneira de exibir a animação do NAM na própria tela do browser para usuários que fiquem receosos em aceitar um arquivo de extensão bat para processamento de comandos em lote;
- d) Criar um login de acesso para armazenar de forma organizada os arquivos gerados pelos usuários;

e) Integrar a outras ferramentas livres que sejam de interesse da universidade e com outros projetos desenvolvidos anteriormente.

6. REFERÊNCIAS

[1]- Tanenbaum, A. S. "Computer Networks". Prentice Hall, 2003.

[2]- Park, Joyce e Converse, Tim. PHP a Bíblia. Editora Campus, 2003.

[3]- The NS Manual formerly NS Notes and Documentation. <http://www.isi.edu/nsnam/ns> - Acessado pela última vez em 30 de junho de 2009.

[4]- Mark Greis Tutorial. <http://www.isi.edu/nsnam/ns/tutorial/index.html>. Acessado pela última vez em 30 de junho de 2009.

[5]- PHP Manual online. <http://www.php.net/docs.php> - Acessado pela última vez em 30 de junho de 2009.

[6]- Apache HTTP Server Project. <http://httpd.apache.org> - Acessado pela última vez em 30 de junho de 2009.

[7]- Coutinho, Mauro Margalho, NETWORK SIMULATOR, Guia Básico para Iniciantes.

[8]- James F. Kurose and Keith W. Ross, “Redes de computadores e a Internet, uma abordagem top-down”. Pearson Education do Brasil, 2006.

[9] Douglas E. Comer, “Internetworking with TCP/IP Principles, Protocols, and Architectures”. Fourth Edition, Prentice Hall.

[10] DARPA (Defense Advanced Research Projects Agency). <http://www.darpa.mil> - Acessado pela última vez em 30 de junho de 2009.

- [11] SAMAN (Simulation Augmented by Measurement and Analysis for Networks). <http://www.isi.edu/saman/index.html> - Acessado pela última vez em 30 de junho de 2009.
- [12] CONSER (Collaborative Simulation for Education and Research). <http://www.isi.edu/conser/index.html> - Acessado pela última vez em 30 de junho de 2009.
- [13] ICIR (The International Computer Science Institute Center for Internet Research). Acessado em <http://www.icir.org> - Acessado pela última vez em 30 de junho de 2009.
- [14] Chin, Liou Kuo, “Rede Privada Virtual – VPN”. <http://www.rnp.br/newsgen/9811/vpn.html> - Acessado pela última vez em 30 de junho de 2009.
- [15] Madeira, Frederico, “VLAN”. <http://www.madeira.eng.br/wiki/index.php?page=VLAN> - Acessado pela última vez em 30 de junho de 2009.
- [16] LAN, MAN, WAN, WPAN, WLAN, WMAN, WWAN, WLL. <http://pt.wikipedia.org> - Acessado pela última vez em 30 de junho de 2009.
- [17] Silva, Adailton J. S., “As Tecnologias de Redes Wireless”. <http://www.rnp.br/newsgen/9805/wireless.html> - Acessado pela última vez em 30 de junho de 2009.
- [18] WPAN. <http://pt.kioskea.net/contents/wireless/wpan.php3> - Acessado pela última vez em 30 de junho de 2009.
- [19] Pinheiro, José Maurício Santos, “A Tecnologia WLL”. http://www.projetoderedes.com.br/artigos/artigo_tecnologia_wll.php - Acessado pela última vez em 30 de junho de 2009.

[20] Monteiro, Alberto Hugo Martins Tavares; Abe, Hugo Andrade e Ferreira, Rômulo Robledo Ferreira, “Proposta de Uso de Redes Mesh em Áreas Periféricas”, Universidade da Amazônia. Acessado pela última vez em 30 de junho de 2009

7. APÊNDICE A – ARQUIVO TCL

```
#Create a simulator object
set ns [new Simulator]

$ns color 1 Blue
$ns color 2 Red
$ns color 3 Green
$ns color 4 Orange

#Open the nam trace file
set nf [open teste.nam w]
$ns namtrace-all $nf

#Define a finish procedure
proc finish {} {
    global ns nf
    $ns flush-trace
    #Close the trace file
    close $nf
    #Execute nam on the trace file
    exec /home/igorcap/ns-allinone-2.33/nam-1.13/nam
    /var/www/teste.nam &
    exit 0
}
```



```

#Declaracao dos nos
set Brasilia [$ns node]
$ns at 0.0 "$Brasilia label Brasilia"
set Curitiba [$ns node]
$ns at 0.0 "$Curitiba label Curitiba"
set Recife [$ns node]
$ns at 0.0 "$Recife label Recife"
set Rio [$ns node]
$ns at 0.0 "$Rio label Rio"
set Saopaulo [$ns node]
$ns at 0.0 "$Saopaulo label Saopaulo"

#Declaracao dos enlaces
$ns simplex-link $Brasilia $Curitiba 1Mb 2ms DropTail
$ns simplex-link $Brasilia $Recife 1Mb 2ms DropTail
$ns simplex-link $Brasilia $Rio 1Mb 2ms DropTail
$ns duplex-link $Rio $Saopaulo 10Mb 1ms DropTail

#Criacao dos agentes
set udp1 [new Agent/UDP]
$udp1 set class_ 1
$ns attach-agent $Brasilia $udp1
set cbr1 [new Application/Traffic/CBR]
$cbr1 set packetSize_ 500
$cbr1 set interval_ 0.005
$cbr1 attach-agent $udp1
set null1 [new Agent/Null]
$ns attach-agent $Curitiba $null1

```

```

$ns connect $udp1 $null1
$ns at 0.1 "$cbr1 start"
$ns at 4.9 "$cbr1 stop"

set udp2 [new Agent/UDP]
$udp2 set class_ 2
$ns attach-agent $Brasilia $udp2
set cbr2 [new Application/Traffic/CBR]
$cbr2 set packetSize_ 500
$cbr2 set interval_ 0.005
$cbr2 attach-agent $udp2
set null2 [new Agent/Null]
$ns attach-agent $Recife $null2
$ns connect $udp2 $null2
$ns at 0.5 "$cbr2 start"
$ns at 4.5 "$cbr2 stop"

set udp3 [new Agent/UDP]
$ns attach-agent $Brasilia $udp3
$udp3 set class_ 3
set cbr3 [new Application/Traffic/CBR]
$cbr3 set packetSize_ 500
$cbr3 set interval_ 0.005
$cbr3 attach-agent $udp3
set null3 [new Agent/Null]
$ns attach-agent $Rio $null3
$ns connect $udp3 $null3
$ns at 0.1 "$cbr3 start"
$ns at 4.5 "$cbr3 stop"

```

```
set udp4 [new Agent/UDP]
$udp4 set class_ 4
$ns attach-agent $Rio $udp4
set cbr4 [new Application/Traffic/CBR]
$cbr4 set packetSize_ 500
$cbr4 set interval_ 0.005
$cbr4 attach-agent $udp4
set null4 [new Agent/Null]
$ns attach-agent $Saopaulo $null4
$ns connect $udp4 $null4
$ns at 1 "$cbr4 start"
$ns at 5 "$cbr4 stop"

#Coloca etiquetas para NAM

#Call the finish procedure after 5 seconds of simulation time
$ns at 5 "finish"

#Run the simulation
$ns run
```