



UNIVERSIDADE FEDERAL DO PARÁ
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
DEPARTAMENTO DE INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

OUTROS TRABALHOS EM:
www.projetoderedes.com.br

Billy Anderson Pinheiro
Cleberson Malcher Guerreiro

ANÁLISE DA VIABILIDADE DE TRANSMISSÃO DE VOZ SOBRE IP EM REDES EM MALHA SEM FIO

BELÉM
2007



UNIVERSIDADE FEDERAL DO PARÁ
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
DEPARTAMENTO DE INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Billy Anderson Pinheiro
Cleberson Malcher Guerreiro

ANÁLISE DA VIABILIDADE DE TRANSMISSÃO DE VOZ SOBRE IP EM REDES EM MALHA SEM FIO

Trabalho de Conclusão de Curso apresentado ao
colegiado do curso de graduação em Ciência da
Computação e ao Departamento de Informática da
Universidade Federal do Pará como condição para
obtenção do título de Bacharel em Ciência da
Computação.

Orientador: Prof. Dr. Antônio Jorge Gomes
Abelém

BELÉM
2007



UNIVERSIDADE FEDERAL DO PARÁ
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
DEPARTAMENTO DE INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Billy Anderson Pinheiro
Cleberson Malcher Guerreiro

ANÁLISE DA VIABILIDADE DE TRANSMISSÃO DE VOZ SOBRE IP EM REDES EM MALHA SEM FIO

Trabalho de Conclusão de Curso apresentado para obtenção do título de
Bacharel em Ciência da Computação.

Data da Defesa:

Conceito: _____

Banca Examinadora

Prof. Dr. Antônio Jorge Gomes Abelém
Depto. de Informática/UFPA – Orientador

Prof. Dr. Kelvin Lopes Dias
Depto. de Engenharia Elétrica e Computação/UFPA - Membro

Prof. Msc. Elisangela Santana Aguiar
UNAMA - Membro

BELÉM
2007

AGRADECIMENTOS

Nomes são apenas rótulos, e o uso de rótulos foi extremamente desaconselhado durante todo o curso, então espero que todos entendam a falta de nomes nesse agradecimento. Mesmo sem nomes, gostaria de agradecer a todos os amigos, professores, desconhecidos, desenhos, livros, filmes e etc. que me ajudaram até aqui, algumas de forma intencional, outras nem tanto, porém sempre me deram força para dar um passo à frente e seguir meu caminho. Também quero agradecer de forma especial a minha família, pelos altos e baixos que passamos, mas por sempre ser uma família. Sem eles eu não existiria, literalmente falando. Enfim, não poderia e nem devo, deixar de agradecer a mim mesmo por ter me deixado terminar esse trabalho, que é apenas um dos muitos moinhos que terei que derrotar nos meus loucos, loucos, loucos sonhos :-).

Billy Anderson Pinheiro

A Deus, por ter me dado saúde para iniciar e concluir este trabalho. Aos meus pais, Adelmir de Oliveira Guerreiro e Dalva Malcher Guerreiro, que durante toda a minha vida escolar estiveram sempre ao meu lado com o intuito de construir um homem digno, vencedor e, acima de tudo, humilde. Ao meu irmão, Cleber Guerreiro, que quando eu tinha que estudar para construir o TCC, sempre me convidava para sair. A toda minha família, por acreditarem e incentivarem os meus estudos. Ao professor Antônio Abelém, por sua zelosa orientação acadêmica e pelo continuado interesse no progresso do meu trabalho de pesquisa. Aos companheiros do Grupo de Trabalho Mesh UFPA, Waldir Moreira e Vagner Nascimento, pelos conhecimentos compartilhados e pela inabalável disposição em nos ajudar. Aos meus amigos de turma do CBCC, pela sua amizade durante estes cinco anos. Aos meus companheiros do Informam, Leandro, Kuroki, Alline e Eder, pela grande convivência que tive no dia a dia e por sua enorme amizade. Ao professor e mestre Arnaldo Prado, pelos seus valiosos ensinamentos e sua eterna amizade, além de suas brincadeiras hilárias. Ao Informam, pelo apoio enquanto bolsista. À minha namorada, Thacyana Souza, pelo seu amor e que, mesmo ela distante, sempre estava presente ao meu lado quando eu mais precisava dela nos meus pensamentos.

Cleberson Malcher Guerreiro.

RESUMO

A inclusão das redes sem fio no mercado e no cotidiano das pessoas está se tornando uma realidade, principalmente como uma alternativa as redes cabeadas quanto à conectividade. Isto é consequência da nova fase que o homem vem atravessando neste mundo globalizado, em que, segundo a literatura, exige facilidades como mobilidade, agilidade e liberdade. Entretanto, a adoção de tecnologias sem fio envolve questões significativas a serem consideradas como, por exemplo, a baixa área de cobertura oferecida pelos seus equipamentos. Isso gerou a criação de novos equipamentos que possam suprir tal necessidade, porém de custo elevado.

A adoção de redes em malha sem fio (*Wireless Mesh Networks – WMN*) cria uma boa alternativa, devido sua viabilidade econômica e expansibilidade. Este trabalho apresenta um estudo de viabilidade de transmissão de voz sobre IP em redes *Mesh* utilizando equipamentos sem fio com *firmware* modificado, baseados em linux. O estudo também aborda questões sobre a qualidade de voz durante a transmissão VoIP na rede *Mesh*.

Palavras-chave: Redes em Malha (Mesh), VoIP, Redes Wireless.

ABSTRACT

The inclusion of wireless networks in the market and in the daily life of the people is becoming a reality, mainly as an alternative to the wired networks because of the connectivity wireless networks provide. This is a consequence of the new phase that man is experiencing in this globalized world, in which, according to the literature, demands things such as mobility, agility and freedom. However, the adoption of wireless technologies involves significant topics to be considered such as, for example, the small coverage area offered by its equipments. The result is the creation of new equipments that can fulfill the needs, however with high costs.

The adoption of wireless mesh networks (WMN) gives a good alternative due to its economical viability and expansibility. This work presents a feasibility study of voice over IP transmission in Mesh networks using wireless equipments (*Wi-Fi*) with modified firmware, based on linux. The study also treats topics in regards to the voice quality during the VoIP transmission in the Mesh network.

Keywords: Mesh Networks, VoIP, Wireless Networks.

SUMÁRIO

1.	INTRODUÇÃO	1
1.1.	OBJETIVOS	2
1.2.	ORGANIZAÇÃO DO TRABALHO	2
2.	REDES EM MALHA (MESH).....	3
2.1.	HISTÓRICO	3
2.2.	CONCEITO	3
2.3.	ARQUITETURA	4
2.4.	PADRÕES MESH	6
2.5.	VANTAGENS E DESVANTAGENS	7
2.5.1	Vantagens.....	7
2.5.2	Desvantagens	8
2.6.	MESH EM DISPOSITIVOS WI-FI	9
2.7.	O PADRÃO IEEE 802.11	9
2.8.	OPENWRT	10
2.9.	OLSR	10
3.	VOIP	12
3.1.	CODECS	13
3.1.1	G.711.....	14
3.1.2	G.729.....	14
3.1.3	GSM.....	15
3.2.	DISPOSITIVOS FÍSICOS PARA VOIP	15
3.3.	VOIP EM WI-FI.....	16
3.4.	VOIP VS. TELEFONIA IP	17
3.5.	CONVERSÃO DE VOZ	18
3.6.	ARQUITETURA VOIP.....	19
3.7.	PROTOCOLOS DE COMUNICAÇÃO VOIP	21
3.8.	O PROTOCOLO RTP	22
4.	MÉTRICAS DE QUALIDADE PARA VOIP SOBRE REDES MESH.....	25
4.1.	QOS EM VOIP	25
4.2.	BANDA	27
4.3.	PERDA DE PACOTES	27
4.4.	TAXA DE JITTER	28
4.5.	RETARDO	30
4.6.	MOS.....	31
5.	ESTUDO DE CASO	33
5.1.	EQUIPAMENTOS	33
5.2.	FERRAMENTAS UTILIZADAS	34
5.3.	CENÁRIO.....	35
5.4.	EXPERIMENTOS	38
5.4.1	Primeiro Experimento.....	38
5.4.2	Segundo Experimento	39
5.5.	RESULTADOS	40
5.6.	RESULTADO DO EXPERIMENTO UM	40
5.7.	RESULTADO DO EXPERIMENTO DOIS	40
6.	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS	45
6.1.	CONSIDERAÇÕES FINAIS	45

6.2. TRABALHOS FUTUROS	45
REFERÊNCIAS	46

LISTA DE FIGURAS

Figura 2.1 - Arquitetura Cliente.	5
Figura 2.2 - Arquitetura Infra-estruturada.	5
Figura 2.3 - Arquitetura Híbrida.	6
Figura 3.4 - realiza chamadas para a rede pública.	18
Figura 3.5 - realiza e recebe chamadas para rede pública.	18
Figura 3.6 - A arquitetura em níveis do VoIP.	20
Figura 4.7 - Relação entre MOS e o Modelo E.	32
Figura 5.8 - Router WRT54GS v4.	33
Figura 5.9 - Cliente SIP : Xten-lite.....	35
Figura 5.11 - Topologia com txpower igual a 19 db.	37
Figura 5.12 - Topologia lógica com 1db de potência.	38
Figura 5.13 - Valores de <i>jitter</i> para uma chamada com duração de 10s utilizando o G.711....	41
Figura 5.14 - Valores de <i>jitter</i> para dez chamadas com duração de 10s utilizando o G.711. ..	42
Figura 5.15 - Valores de <i>jitter</i> para vinte chamadas com duração de 10s utilizando o G.711.	42
Figura 5.16 - Valores de <i>jitter</i> para quarenta chamadas com duração de 10s utilizando o G.711.	43
Figura 5.17 - Número de chamadas em função da perda para o G.711.....	44

LISTA DE TABELAS

Tabela 3.1 - Comparação entre características da Telefonia Convencional e o Serviço VoIP	12
Tabela 3.2 - Especificações dos padrões de codificação do sinal de voz PCM e ADPCM, além de LD-CELP, CS-ACELP, MP-MLQ e ACELP.....	19
Tabela 4.3 - [TANENBAUM, 2003] - Aplicações e seus requisitos de QoS.....	26
Tabela 4.4 [DIGITAL AUDIO, 2007] - Pontuação MOS.....	31
Tabela 4.5 [CISCO, 2006] - Pontuação MOS de alguns CODEC's.	31
Tabela 5.6 - Relação IPs X Roteadores	36

LISTA DE SIGLAS

ACELP	Algebraic Code Excited Linear Prediction
ADPCM	Adaptive Differential Pulse Code Modulation
ADSL	Asymmetric Digital Subscriber Line
AP	Access Point
ARPANET	Advanced Research Projects Agency Network
ASN1	Abstract Syntax Notation One
ATA	Analog Telephone Adapter
BSS	Basic Service Set
CNG	Comfort Noise Generation
CPU	Central Processing Unit
cRTP	compressed Real-time Transport Protocol
CS-ACELP	Conjugate-Structure Algebraic-Code-Excited Linear-Prediction
DARPA	Defense Advanced Research Projects Agency
DHCP	Dynamic Host Configuration Protocol
DS	Distribution System
DSP	Digital Signal Processor
ESS	Extended Service Set
ETX	Expected Transmission Count
GPL	GNU Public License
GSM	Global System for Mobile Communications
HTTP	HyperText Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IMTC	International Multimedia Teleconferencing Consortium
IP	Internet Protocol
IPSec	IP Security
ISO	International Organization for Standardization
ISP	Internet Service Provider
ITU-T	International Telecommunications Union – Telecommunications Standardization Sector
LAN	Local Area Network
LD-CELP	Low-Delay Code Excited Linear Prediction
LPAD	Lightweight Directory Access Protocol
MOS	Mean Opinion Score
MP-MLQ	Multi-Pulse, Multi-Level Quantization
MPR	Multi-Point Relays
NVRAM	Non-Volatile Random Access Memory
OLSR	Optimized Link State Routing
PABX	Private Automatic Branch Exchange
PCM	Pulse Code Modulation
PDA	Personal Digital Assistant
PER	Packing Encoding Rules
PINT	PSTN-Internet Interworking
PLR/PER	packet error/loss rate
QoS	Quality of Service
RAM	Random Access Memory

RTCP	Real-Time Transport Control Protocol
RTP	Real-Time Transport Protocol
SAP	Service Advertisement Protocol
SIP	Session Initiation Protocol
SMTP	Simple Mail Transfer Protocol
SSH	Secure Shell
TCP	Transmission Control Protocol
UDP	User Datagrama Protocol
VAD	Voice Activity Detection
VoIP	Voice over IP
VoWiFi	VoIP over Wireless Fidelity
VQ	Voice Quality
WAN	Wide Area Network
Wi-Fi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WMN	Wireless Mesh Networks

1. INTRODUÇÃO

As redes sem fio têm a cada momento adquirido mais espaço sobre as redes cabeadas, porém um grande empecilho à adoção de redes sem fio tem sido a baixa área de cobertura oferecida pelos equipamentos. Visando minimizar esses problemas, diversos aparelhos e tecnologias foram criados, no entanto junto com o aumento da área de cobertura crescia também o preço dos equipamentos impossibilitando uma implantação massiva.

As redes em malha sem fio (*Wireless Mesh Networks – WMN*), ou simplesmente redes *mesh*, surgem como uma alternativa barata e com grande poder de abrangência, tornando-as popular para a última milha de acesso. Contudo esta proposta sofre principalmente com a falta de padronização da tecnologia. Diversas empresas como a Motorola e a Cisco já possuem soluções denominadas *pré-mesh*, porém o alto custo destas dificulta sua adoção em larga escala.

Como alternativa às soluções proprietárias, podem ser usados equipamentos *Wi-Fi* (*Wireless Fidelity*) com *firmware* modificado, utilizando uma distribuição linux voltada para dispositivos embarcados, denominada OpenWRT, para operarem como equipamentos *mesh*, permitindo o fácil compartilhamento de recursos da rede bem como diminuindo consideravelmente o custo total de implantação. Este tipo de configuração permite que sejam criadas redes de acesso público a um custo bem acessível, propiciando uma série de serviços aos usuários.

Um dos muitos serviços que podem ser oferecidos através das redes *mesh* é o serviço de voz sobre IP (*Voice over IP – VoIP*), permitindo que os usuários tenham uma comunicação fácil e barata. Diversos são os requisitos que o serviço de VoIP necessita, como por exemplo *jitter* e perda de pacotes, cabendo a estrutura de redes atender aos mesmos. Esta tarefa já é um verdadeiro desafio nas redes tradicionais, sofrendo um grande acréscimo de complexidade nas WMNs.

Este trabalho tem como objetivo mostrar o uso de dispositivos *Wi-Fi* operando como equipamentos *mesh*, bem como, a viabilidade do uso de VoIP sobre uma rede em malha sem fio com o uso do *Asterisk*¹ e o protocolo da iniciação da sessão (*Session Initiation Protocol - SIP*) e a utilização da ferramenta *iperf* para a simulação de tráfego de voz na redes.

1.1. OBJETIVOS

Este trabalho tem como objetivos principais, apresentar um estudo sobre a tecnologia *Mesh*, realizar teste piloto de redes *mesh* usando dispositivo sem fio na UFPA e a verificação de viabilização de tráfego VoIP sobre redes *mesh*.

Como objetivos específicos têm-se um levantamento histórico, conceitual e funcional de redes em malha sem fio; modificação de equipamentos *Wi-Fi* para criação de roteadores *mesh*; uso destes roteadores para implantação de ambientes *mesh*; Geração e simulação de chamadas de voz sobre IP (Internet Protocol); e avaliação de *jitter* e perda para mensurar a qualidade de voz.

1.2. ORGANIZAÇÃO DO TRABALHO

Inicialmente este trabalho realiza um levantamento teórico sobre redes mesh visto no Capítulo 2. Nele encontram-se conceitos e definições, arquiteturas e padronização de redes mesh, além das vantagens e desvantagens na adoção deste tipo de rede. Também neste capítulo, são abordadas questões relacionadas quanto ao uso de dispositivos ou equipamentos sem fio modificados para possibilitar o uso de WNM

O Capítulo 3 trata da tecnologia VoIP, suas características, assim como seu conceito, sua arquitetura de nível, bem como seu uso em ambientes sem fio. Será visto também os protocolos de comunicação mais utilizados em VoIP.

O Capítulo 4 apresenta as métricas de qualidade para VoIP sobre redes mesh. Nela será visto a questão da QoS em VoIP, assim como os parâmetros necessários para uma boa qualidade de voz.

Já no Capítulo 5, será exposto o estudo de caso deste trabalho. Neste capítulo, será visto o cenário, os equipamentos, as ferramentas utilizadas, além dos resultados obtidos durante a realização dos experimentos na rede. Finalmente, o Capítulo 6 trata das considerações finais e os trabalhos futuros para este trabalho.

2. REDES EM MALHA (MESH)

Neste tópico serão tratadas as questões relacionadas às redes *mesh*. Primeiramente será visto um breve histórico, em seguida, um conceito geral sobre as redes *mesh* e sua arquitetura. Logo após, serão apresentados os padrões para esta rede e, finalmente, as vantagens e as desvantagens do seu uso.

2.1. HISTÓRICO

A primeira implementação de redes *Mesh* foi feita pelo ARPANET (*Advanced Research Projects Agency Network*), sendo esta baseada no trabalho de Paul Baran no início dos anos 1960. Esta implementação gerou vários dos algoritmos de roteamento utilizados na Internet atualmente. No entanto a maioria das técnicas realmente usadas em redes *mesh* foi criada nos anos 1970 e 1980 pelo DARPA (Defense Advanced Research Projects Agency) [BREUEL, 2004] com o objetivo de buscar uma rede que permitisse a fácil comunicação entre cada componente (soldados, tanques, helicópteros e afins) em um campo de batalha sem a necessidade de um ponto central.

2.2. CONCEITO

Como um tópico em pleno desenvolvimento, sem uma padronização existente e com várias implementações diferentes, ainda existem dúvidas a respeito do conceito de redes *mesh*. De maneira geral e simplificada, uma rede *mesh* é um caso especial de rede *ad hoc* [LUIZ E JÚNIOR, 2005]. Estas são definidas por Ramanathan [RAMANATHAN e REDI, 2002] como sendo um conjunto de dispositivos que não possuem uma infra-estrutura fixa nem uma organização prévia dos enlaces de comunicação, sendo cada dispositivo responsável por descobrir as rotas de acesso direto a seus vizinhos formando, com isso, uma rede de múltiplos saltos. Cabe ainda ressaltar a existência de mobilidade entre os nós que, juntamente com a possível entrada e saída destes na rede, podem provocar o surgimento de múltiplas rotas entre dois pontos de acesso [ALBUQUERQUE, 2006], sendo que a adição de mais nós na rede aumenta sua complexidade.

As WMNs são redes de múltiplos saltos, auto-organizáveis e auto-configuráveis que podem apresentar nós roteadores, que possuem posições fixas e têm a capacidade de fazer o papel de *gateway* ou *bridge* além de apresentarem múltiplos saltos e nós clientes, que podem

ter mobilidade, porém apenas apresentam a característica de múltiplos saltos [AKYILDIZ et al, 2007].

2.3. ARQUITETURA

Dois tipos de nós podem ser encontrados nas redes *mesh*, no entanto as redes *mesh* aceitam a comunicação com outros tipos de redes e seus respectivos equipamentos.

- **Roteadores *Mesh* (*Mesh Routers*):** Possuem as mesmas funcionalidades de roteadores convencionais, como *gateway* e *bridge*, porém com o suporte a *mesh*, que provê maior flexibilidade a rede, pois permite a comunicação com outros tipos de redes, como a cabeada, através do uso de múltiplas interfaces. Para que todos estes recursos possam ser executados de maneira satisfatória é necessário um maior poder computacional, necessitando normalmente de um computador para realizar o papel de nó central ou simplesmente fazer uso de um sistema embarcado [AKYILDIZ et al, 2007].
- **Clientes *Mesh* (*Mesh Clients*):** Podem realizar o processo de encaminhamento de pacotes entre os demais elementos *mesh* da rede, no entanto não podem exercer as funções de *bridge* ou *gateway*. Em contrapartida, este tipo de nó apresenta apenas uma interface de rede e um hardware bem mais simples, podendo variar desde um laptop até um telefone IP [AKYILDIZ et al, 2007].

Através da combinação desses dois tipos de nós *mesh* juntamente com a utilização de interfaces de redes não *mesh*, podem ser formadas três tipos de arquiteturas:

- **Arquitetura Cliente:** Apenas nós clientes (*Mesh Clients*) são usados nesta arquitetura. Cada nó faz tanto o papel de cliente como o de roteador, como mostra a Figura 2.1. Eles se comunicam como em uma rede *peer-to-peer*² formando uma estrutura muito próxima a de uma rede *Ad Hoc*, diferindo apenas na utilização de uma única tecnologia de transmissão [AKYILDIZ et al, 2007].

² É um tipo de rede onde os nós possuem funcionalidades equivalentes, ou seja, todos fazem o papel de cliente e de servidor.

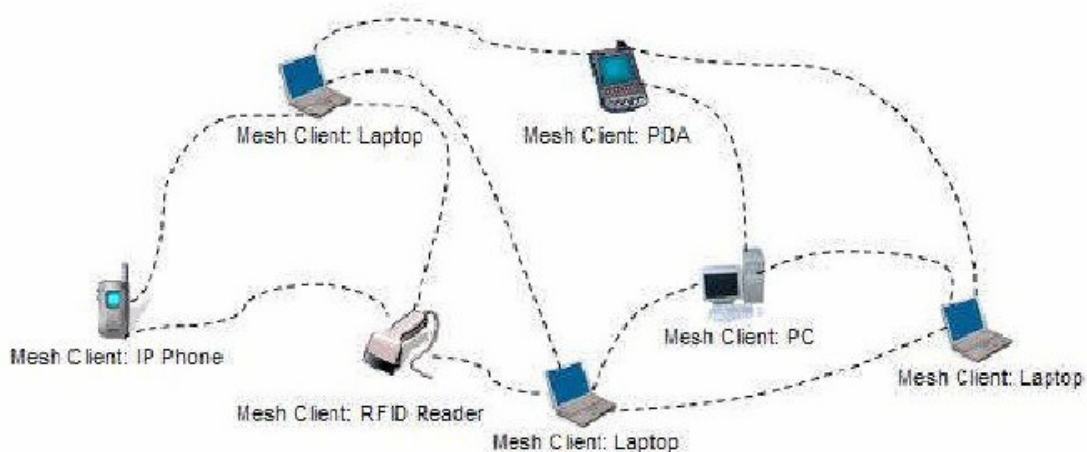


Figura 2.1 [AKYILDIZ et al, 2007] - Arquitetura Cliente.

- **Arquitetura Infra-Estruturada:** O *backbone* da rede é composto de roteadores *mesh* que fornecem a infra-estrutura básica para a conexão de clientes não *mesh*, vista na Figura 2.2. Através deste *backbone* formado é possível interligar diferentes redes com diferentes tecnologias de transmissão. Esse é o tipo de rede *mesh* mais usada, pois necessita de modificações apenas nos seus roteadores que normalmente utilizam duas antenas com canais distintos, uma para o *backbone* e outra para atender os clientes [AKYILDIZ et al, 2007].

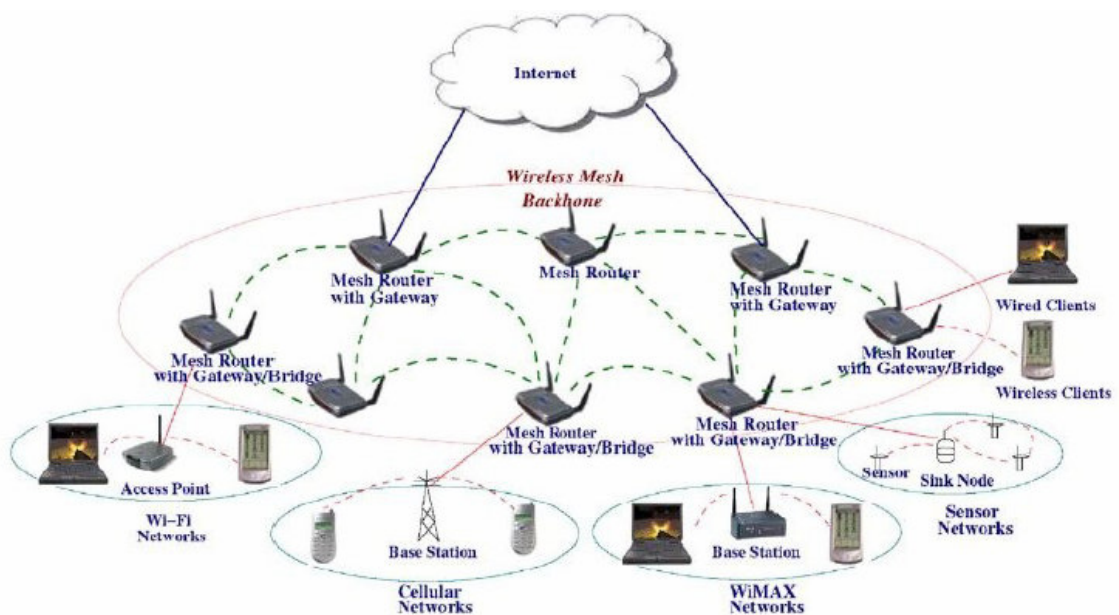


Figura 2.2 [AKYILDIZ et al, 2007] - Arquitetura Infra-estruturada.

- **Arquitetura Híbrida:** Esta arquitetura faz o uso tanto dos roteadores como dos clientes *mesh*. Ela é a configuração mais completa, fazendo o uso de todas as possibilidades de comunicações que as redes *mesh* oferecem, possibilitando que clientes *mesh* e convencionais tenham acesso ao *backbone mesh* que oferece uma série de interligações com outras redes [AKYILDIZ et al, 2007]. A Figura 2.3 apresenta este tipo de arquitetura.

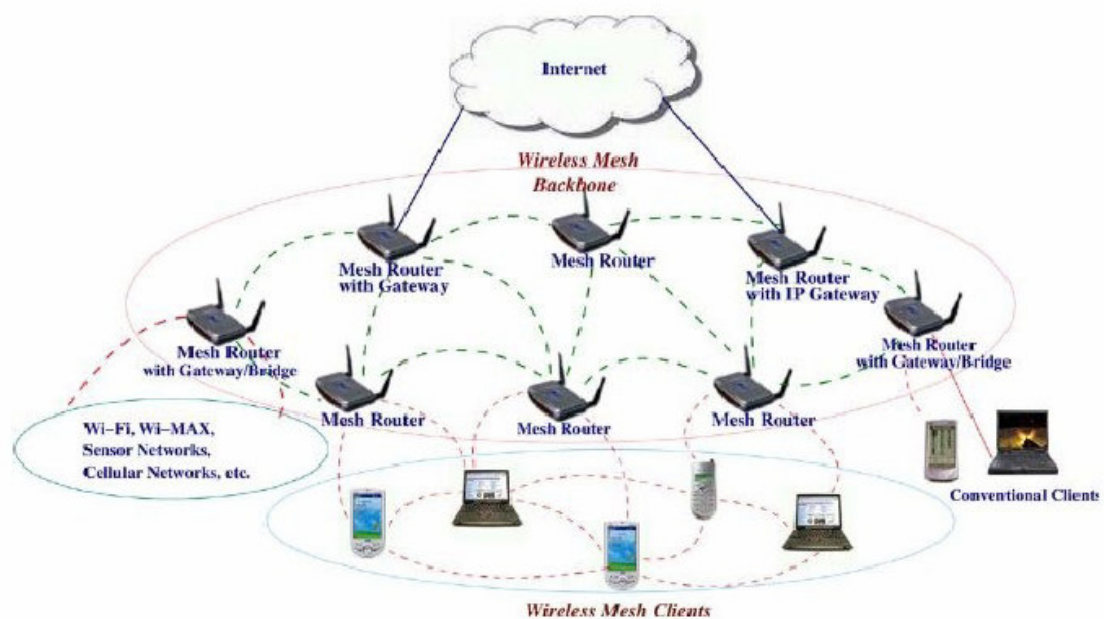


Figura 2.3 [AKYILDIZ et al, 2007] - Arquitetura Híbrida.

Das arquiteturas expostas cada uma tem seu grau de utilização e aplicação, cabe lembrar que as WMNs, não estão restritas a tecnologia *Wi-Fi*, sendo possível a utilização, por exemplo, de dispositivos *Bluetooth*³ em uma arquitetura Cliente *Mesh*, assim como vários outros tipos de configurações.

2.4. PADRÕES MESH

Saindo na frente do Instituto dos Engenheiros Eletricistas e Eletrônicos (*Institute of Electrical and Electronics Engineers - IEEE*), vários fabricantes já lançaram no mercado um grande numero de equipamentos denominados de *Pré-Mesh*, porem estas soluções proprietárias apresentam um custo muito alto, além de não fornecer qualquer garantia de

3 www.bluetooth.com

compatibilidade com os aparelhos de outros fabricantes [BRASIL, 2006].

Em Julho de 2005 foi iniciado o processo de padronização da tecnologia *mesh*, 15 propostas foram feitas por diferentes empresas e grupos. Apenas em Janeiro de 2006 houve um consenso e as propostas dos grupos *SEEMesh* e *Wi-Mesh Alliance* foram unidas e deram origem ao 802.11s que tem previsão de termino em 2008.

Cabe ressaltar o quanto se espera dessa tecnologia e, por conseguinte, o quanto se espera dessa padronização, a possibilidade de interligar diferentes redes, com diferentes tecnologias de transmissão, de uma maneira simples e barata torna o sonho de ter cidades totalmente cobertas por redes sem fio cada vez mais real [TASK GROUP S, 2007].

Um exemplo é o uso do padrão de Interoperabilidade Mundial para Acesso de Micro-ondas (*Worldwide Interoperability for Microwave Access - WiMax*) em conjunto com redes *mesh* que torna possível a criação de redes *mesh* de WiMax fixo, cobrindo inteiramente regiões metropolitanas com banda larga sem fio, não sendo preciso a compra de equipamentos WiMax na ultima milha [PESSOA et al, 2007].

2.5. VANTAGENS E DESVANTAGENS

O conceito de redes *mesh* traz consigo uma serie de vantagens que tornam cada vez mais interessantes a sua implantação, porém existem algumas desvantagens, a maioria presentes ainda pela recente atenção dada as redes *mesh* por parte tanto do mercado quanto da academia.

2.5.1 Vantagens

- Redes de Baixo custo: O compartilhamento de recursos faz com que o custo total da rede caia, viabilizando a criação de redes comunitárias [LUIZ E JÚNIOR, 2005].
- Fácil Implantação: Como as redes *mesh* possuem a característica de serem auto-configuráveis, a sua implantação se torna fácil, pois não são necessárias configurações complexas, nem necessidade de mudança caso algum nó venha a entrar na rede.
- Tolerante a falhas: A capacidade de roteamento dinâmico aliado à existência de múltiplas rotas de acesso a um nó faz com que a rede consiga se recuperar de falhas como a perda de um enlace de comunicação.
- Escalável: Uma das melhores características das redes *mesh* é que sua capacidade de

roteamento cresce conforme os nós são adicionados, logo o crescimento das redes, diferente da arquitetura tradicional não é um problema [HARADA, 2006].

2.5.2 Desvantagens

- Falta de Padronização: Este problema impossibilita até então a adoção da tecnologia em larga escala, espera-se que em 2008 tal problema já esteja solucionado [STATUS, 2007].
- Alto preço dos aparelhos: Atualmente o preço dos aparelhos torna o acesso a estes muito restritivos, espera-se que com a padronização da tecnologia os preços tornem-se mais acessíveis.
- Interferência: O uso da faixa, não regulamentada de 2.4 GHz, possibilita a interferência de equipamentos externos à rede que degradam a qualidade desta como um todo [STATUS, 2007].
- Baixo *throughput*: Os valores atuais ainda precisam ser aprimorados, tendo quem mente que a possibilidade de crescimento de uma rede *mesh*. Uma alternativa para maximizar o *throughput* é utilizar um canal exclusivo para o tráfego de *backhaul*⁴, o que gera um desempenho bastante superior [KYASANUR, 2007].
- Falta de segurança: A segurança ainda é um campo aberto no que diz respeito às redes *mesh*, além dos problemas normais de segurança em redes sem fio, ainda existe o problema de garantir a privacidade dos dados que estão trafegando entre os nós [BREUEL, 2004].
- Ausência de Qualidade de Serviço: Assim como os problemas de segurança, a falta de qualidade de serviço em redes *mesh* é um campo de pesquisa muito pouco explorado, tendo algumas propostas para estes problemas, como o algoritmo de roteamento denominado de roteamento de redes *mesh* sem fio (*Wireless Mesh Routing - WMR*) [XUE e GANZ, 2005].

4 Transmitir dados para um *backbone* de rede.

2.6. MESH EM DISPOSITIVOS WI-FI

Como as atuais soluções pré-*mesh* são proprietárias, caras e na maioria das vezes não interoperáveis, uma solução alternativa usando equipamentos *Wi-Fi* e software livre foi desenvolvida. Nesta solução ocorre a modificação nos *firmwares* dos equipamentos *Wi-Fi*, através da adoção de linux embarcado, com o protocolo de roteamento alterado para atender as necessidades de uma rede *mesh*.

A primeira experiência em redes *mesh* usando esta solução no Brasil foi realizada pelo projeto *Remesh*⁵ da Universidade Federal Fluminense (UFF), onde uma rede de acesso comunitário foi implantada nos arredores do campus universitário da praia vermelha, sendo que um protótipo de um roteador *mesh* foi criado pelo projeto para possibilitar a implantação [ALBUQUERQUE, 2006].

2.7. O PADRÃO IEEE 802.11

Conhecido popularmente como *Wi-Fi*, o padrão IEEE 802.11 [IEEE,1999], projetado pelo IEEE como padrão para redes locais sem fio (*Wireless Local Area Network - WLAN*), tornou-se um dos mais famosos padrões nas comunicações *wireless* entre equipamentos de computação e acessórios. O padrão 802.11 permite dois modos para o seu funcionamento. Um deles caracterizado pela presença de um ponto que controla as transmissões das estações (infra-estruturada) e o outro apenas com as estações interligadas entre si (*ad hoc*). Neste trabalho não é focado o estudo sobre este padrão, contudo, seu uso é fundamental na atuação dos dispositivos *Wi-Fi* da rede a ser analisada.

A arquitetura do padrão 802.11 define elementos fundamentais para oferecer suporte à mobilidade de estações de forma transparente para as camadas superiores. Em redes do tipo infra-estruturadas, a arquitetura é agrupada em um bloco fundamental de construção definido como conjunto básico de serviços (*Basic Service Set – BSS*).

De acordo com o padrão, o BSS é composto por um determinado grupo de estações sem fio controladas por um ponto de acesso (*Access Point - AP*) que fornece conectividade com as camadas acima. É possível criar uma rede infra-estruturada que possa conter inúmeros pontos de acesso interligados por um sistema de distribuição (*Distribution System - DS*),

5 <http://mesh.ic.uff.br/>

podendo ser composto por um *backbone*⁶. O DS tem como finalidade aumentar a cobertura da rede caso tenha um cenário onde existam vários sistemas de distribuição assim como seus pontos de acesso. Neste caso, aplica-se a este cenário a denominação de conjunto estendido de serviços (*Extended Service Set - ESS*). Nele, é permitido que as estações transitem de um BSS para outro, além de que, na presença de um elemento especial chamado *portal*, o ESS será capaz de se comunicar com redes locais que seguem diferentes padrões (802.X).

2.8. OPENWRT

O *firmware* dos equipamentos é modificado e passa a operar com uma distribuição linux chamada OpenWRT [OPENWRT, 2006]. Esta distribuição é fruto do projeto com o mesmo nome que nasceu em Janeiro de 2004. Inicialmente o projeto era baseado no código do roteador WRT54G da Linksys, que adota a Licença Pública GNU (*GNU Public License - GPL*⁷). Posteriormente em 2005 o projeto passou a adotar o linux como o kernel padrão, realizando apenas modificações para que a distribuição passe a dar suporte aos diferentes tipos de interfaces de rede sem fio [WIKI-OPENWRT, 2007].

O objetivo do projeto é fornecer um simples e estático *firmware* que tenha todas as funcionalidades de um sistema linux básico, mais um conjunto de aplicativos que podem facilmente serem instalados no sistema através de um gerenciador de pacotes semelhante ao *apt-get*⁸ chamado *ipkg*. Um grande diferencial do OpenWRT é disponibilizar um sistema de arquivo que permite a leitura e a escrita de forma permanente [WIKI-OPENWRT, 2007].

Alem do OpenWRT existe ainda outros projetos que oferecem um firmware para roteadores sem fio. Um exemplo é o DDWRT⁹ que oferece um firmware muito parecido com o do OpenWRT, porem com uma interface de gerencia web.

A escolha pelo uso do OpenWRT deve-se a sua maturidade, maior numero de pacotes disponíveis para a instalação e pelo menor tamanho do firmware gerado, já que os equipamentos possuem uma limitada capacidade de armazenamento.

2.9. OLSR

Ainda não existe um protocolo desenvolvido especialmente para redes *mesh*, no

6 Link central de distribuição da redes

7 www.gnu.org

8 Gerenciador de pacotes padrão de sistemas Debian, com suporte a resolver dependências.

9 www.dd-wrt.com

entanto devido a similaridades entre as redes *ad hoc* e as redes *mesh*, é comum o uso de protocolos da primeira em redes mesh [ABELÉM, 2007].

O protocolo de roteamento dinâmico mais usado é o protocolo de roteamento de estado de enlace (*Optimized Link State Routing Protocol - OLSR*), uma versão melhorada do clássico algoritmo de estado de enlace, que possui uma natureza pró-ativa, ou seja, o descobrimento das rotas é realizado de tempos em tempos e utiliza o conceito de *MultiPoint Relays* (MPRs) onde os nós MPRs são escolhidos para fazer o encaminhamento dos pacotes de controle durante o processo de descobrimento e manutenção das rotas, minimizando assim o *overhead* na rede [CLAUSENT et al, 2003].

O OLSR usado é o fornecido pelo olsr.org que é um projeto que mantém uma implementação GPL com um acréscimo de uma métrica que trabalha usando como parâmetro a qualidade do link de cada enlace, a contagem esperada de transmissão (*Expected Transmission Count - ETX*) [OLSR DAEMON, 2007].

3. VOIP

A tecnologia VoIP vem sendo uma grande revolução nas telecomunicações, resultante do incrível crescimento das redes baseadas em pacotes, especialmente da Internet [CHOWDHURY, 2002]. Esta revolução está unificando os mundos de dados e telecomunicações em uma só rede convergente ubíqua. Esta tecnologia, sem dúvida, abre um novo horizonte para as possíveis aplicações integrando-se voz e dados em um mesmo equipamento terminal de usuário, aproximando pessoas geograficamente distantes, aumentando a interatividade de aplicativos e diminuindo os custos de comunicação quando comparada às convencionais ligações telefônicas interurbanas.

A tecnologia VoIP é um conjunto de programas e equipamentos usado para transmitir chamadas telefônicas através do envio de dados de voz por meio de pacotes IP, isto é, ela possibilita que pacotes IP sejam utilizados para transportar chamadas de voz ao invés de chamadas oriundas dos circuitos tradicionais da rede pública de telefonia comutada. Primeiramente, o transmissor envia um sinal da voz que é digitalizado e segmentado em um pacote IP, com a possibilidade de ser transmitido junto com outros pacotes, como dados [CHOWDHURY, 2002].

Quando os pacotes chegam ao receptor, eles são remontados e os sons normais de voz são capturados. Isso permite que a tecnologia VoIP seja capaz de oferecer um meio para integrar algumas capacidades de voz e dados, de modo que ambos possam ser transportados pelas redes IP. Esta capacidade cria uma série de vantagens do serviço VoIP em comparação a telefonia convencional [TELECO, 2006]. A Tabela 3.1 mostra algumas características específicas do VoIP e da telefonia convencional.

Tabela 3.1 - Comparação entre características da Telefonia Convencional e o Serviço VoIP

Característica	Telefonia Convencional	Serviço VoIP
Tipo de Conexão	Cabo de cobre (par trançado)	Banda Larga de Internet
Falta de Energia Elétrica	Continua funcional	Pára de funcionar
Mobilidade	Limitada a casa do usuário	Acesso em qualquer lugar do mundo, desde que conectado a Internet
Número Telefônico	Associado ao domicílio do usuário	Associado à área local do número contratado
Chamadas locais	Área local do domicílio do	Área local do número

	usuário	contratado
--	---------	------------

Existe uma variedade de *softwares* que oferecem suporte a esta tecnologia, como Google Talk¹⁰ e o famoso Skype¹¹, cada um com sua peculiaridade, porém com o mesmo objetivo de prover uma comunicação adequada para chamadas telefônicas sobre redes IP. Através destes *softwares*, é possível estabelecer uma chamada oriunda de um computador para uma linha telefônica convencional, ou para outro computador, desde que o mesmo utilize um *software* que tenha suporte à telefonia IP. Mas a grande vantagem, sem dúvida, da tecnologia VoIP é o seu baixo custo por chamada, em comparação com a telefonia comutada, em chamadas de computadores para telefones convencionais. Caso seja para outro computador, o custo da chamada é gratuito.

3.1. CODECS

Como sugere, os Codificadores-Decodificadores (*coder-decoder – codec*), são usados para codificar e decodificar, ou compactar e descompactar, vários tipos de dados, principalmente aqueles que, de outra forma, usariam enormes quantidades de espaço em disco. Os *codecs* comuns incluem aqueles que convertem sinais de som analógico em som digitalizado. Os *codecs* podem ser usados com conteúdo de *streaming* (vídeo e áudio ao vivo) ou conteúdo baseado em arquivo.

Os mecanismos de compressão de voz visam a otimização da utilização da largura de banda para a sua transmissão. Os algoritmos de compressão de voz são padronizados pela União Internacional de Telecomunicações (*International Telecommunications Union – Telecommunications Standardization Sector - ITU-T*), através das recomendações da série G.7xx. Estes mecanismos executados pelos *codec* que nada mais são que circuitos integrados especializados através de um Processador Digital de Sinal (*Digital Signal Processor - DSP*) e dedicados para o processamento dos sinais de voz com sua conversão analógica para a digital.

Cada um dos diversos algoritmos de compressão possui características de desempenho intrínsecas, tais como: atraso de processamento dos sinais de voz (atraso de compressão da voz) e taxa de produção das amostras digitais de voz. Os métodos de compressão de voz mais

¹⁰ www.google.com.br/talk

¹¹ www.skype.com

empregados são: Modulação por Codificação de Pulsos (*Pulse Code Modulation – PCM*), Modulação por Codificação de Pulsos Diferenciais Adaptáveis (*Adaptive Differential Pulse Code Modulation – ADPCM*), *Low-Delay Code Excited Linear Prediction* (LD-CELP), *Conjugate-Structure Algebraic-Code-Excited Linear-Prediction* (CS-ACELP), *Multi-Pulse, Multi-Level Quantization* (MP-MLQ) e a *Algebraic Code Excited Linear Prediction* (ACELP).

De acordo com o número de ciclos de Unidade Central de Processamento (*Central Processing Unit - CPU*) utilizados nos DSP, os *Codecs* para a compressão de voz podem ser classificados como: *Codec* de Alta Complexidade (*High Complexity*) e *Codec* de Média Complexidade (*Medium Complexity*).

Para este trabalho foram utilizados os *codecs* de áudio G.711, G.729 e o GSM 6.10, dadas as suas características com relação ao som produzido [CISCO, 2006].

3.1.1 G.711

A recomendação G.711 da ITU-T (*Telecommunication Standardization Sector of International Telecommunication Union*) é o padrão internacional para codificação de áudio telefônico em um canal de 64 Kbps. Esse é um esquema PCM operando em uma taxa de amostragem de 8 kHz, com 8 bits por amostra. Esta recomendação é a padronização da técnica de PCM.

A frequência de amostragem obedece ao teorema de *Nyquist* para que o sinal possa ser restaurado com perfeição no processo de decodificação. A taxa de saída deste codificador, desconsiderando qualquer *overhead*, é, portanto, 64 kbps. Os codificadores evoluíram a partir do PCM com a finalidade de reduzir cada vez mais a taxa de bits necessária para que se mantenham transmissões de voz com qualidade.

3.1.2 G.729

Esta a recomendação, patenteada pela empresa *VoiceAge Solutions*, é utilizada para compressão de voz e supressão de silêncio de um sinal digital, que consegue converter um sinal PCM de entrada de 64 kbps em uma saída de 8 kbps. Usa pouca banda e possui uma qualidade de áudio impressionante, usando o método de compressão CS-ACELP (*Conjugate-*

Structure Algebraic-Code-Excited Linear-Prediction), a despeito do alto consumo de processamento requerido.

Para a recomendação ITU-T G.729, a implementação de *Codec* de Média Complexidade é identificada [ITU-T, 2007] como G.729a (inserção do código “a”). Já a implementação de *Codec* de Alta Complexidade é identificada apenas por G.729. Caso o *Codec* G.729 (Média ou Alta Complexidade) possua suporte incorporado à implementação do IETF (*Internet Engineering Task Force*) VAD (*Voice Activity Detection*) e do CNG (*Comfort Noise Generation*), o código “b” é utilizado para indicar esta característica.

3.1.3 GSM

Os sistemas celulares incorporaram o *Global System for Mobile Communications* (GSM), que é um *codec* de áudio que não requer licença de uso e oferece um bom desempenho, que opera a 13 kbps, através do uso do *Regular Pulse Excited* (RPE), embora não tão boa quanto o G.729. Sua vantagem principal sobre outros codecs baixos da taxa é sua relativa simplicidade, pois opera facilmente em tempo real e permite o uso de máquinas que não possuem um alto poder de processamento, visto que os *codecs* LD-CELP e CS-ACELP necessitam um DSP dedicado.

3.2. DISPOSITIVOS FÍSICOS PARA VOIP

Para criar um ambiente VoIP que esteja adequado para uma boa conversação sem a utilização de computadores, é necessário a presença de equipamentos especiais que são exclusivos neste cenário. Caso o cenário esteja usando o computador como um dos terminais na comunicação VoIP, então o computador será usado como telefone IP, desde que tenha uma placa de som, um microfone, alto falantes ou fones de ouvidos, e um programa do tipo *softphone*, que possui todos os recursos para funcionar como um telefone IP.

Sabe-se que os serviços VoIP utilizam telefones apropriados para as redes IP, e que são muito diferentes, em complexidade, dos telefones analógicos convencionais. Isso se dá por serem digitais e possuírem recursos semelhantes àqueles encontrados nos computadores. Primeiramente, é preciso um Adaptador para Telefone Analógico (*Analog Telephone Adapter - ATA*) [CHOWDHURY, 2002]. Este dispositivo funciona como um conversor de telefone IP

para um telefone analógico convencional. O ATA é conectado a um acesso de banda larga (rede IP) e a um telefone analógico convencional, que pode ser usado normalmente para fazer e receber ligações do serviço VoIP contratado. Além do ATA, utiliza-se o Telefone IP, que é um telefone que possui todos os recursos necessários para um serviço VoIP. Para ser usado é necessário apenas conectá-lo a um acesso de banda larga (rede IP) para fazer e receber ligações do serviço VoIP.

Para um ambiente mais externo, verifica-se a presença de provedores de Internet (*Internet Service Provider - ISP*), além de um servidor VoIP. Estes provedores de Internet disponibilizam o acesso a Internet na comunicação. Ainda assim, um ambiente completo de VoIP pode incluir outros elementos como *gateways*, *firewalls*, roteadores e protocolos.

3.3. VOIP EM WI-FI

Muitas são as facilidades que chegaram às comunicações com a entrada da tecnologia VoIP. Recentemente, com a crescente aceitação de tecnologias sem fio, em especial do *Wireless Fidelity (Wi-Fi)*, o serviço VoIP incorporou este tipo de comunicação, gerando o que chamamos de Voz sobre redes sem fio (*VoIP over Wireless Fidelity - VoWiFi*) [PERCY e HOMMER, 2003].

VoWiFi é a versão *wireless* desta tecnologia que é projetada trabalhar em dispositivos *wireless* tais como um *laptop* ou um Assistente Pessoal Digital (*Personal Digital Assistant – PDA*). Esta adoção da tecnologia VoIP à comunicação sem fio deve-se ao fato de que empresas e organizações enxergaram uma forma de reduzir significativamente os custos econômicos gerados pela telefonia móvel, além de oferecer melhor qualidade de voz (*Voice Quality - VQ*) e maior cobertura para suas comunicações móveis. Os sistemas *wireless* de VoIP proprietários têm sido usados por muito tempo nas empresas, nos varejistas e nas organizações de Tecnologia da Informação, mas hoje, estas organizações e as instituições educacionais estão adotando cada vez mais a voz sobre soluções *Wi-Fi*, com a introdução de padrões da indústria, tais como o SIP para um melhor controle da chamada [BENCHIMOL et al, 2005].

Apesar das vantagens que o *VoWiFi* pode oferecer, ele ainda é carente em algumas características importantes para que sua transmissão seja competitiva com a telefonia celular. As mais relevantes se referem às questões de segurança e transição limitada, fundamentais na

comunicação sem fio. Com o avanço das pesquisas neste tipo de tecnologia, é provável que soluções sejam encontradas para estas questões, o mais rápido possível.

3.4. VOIP VS. TELEFONIA IP

O tratamento para o conceito de VoIP geralmente é associado ao conceito de Telefonia IP, entretanto, cada um possui uma definição sua própria. Como foi dito, VoIP corresponde a comunicação de voz sobre redes IP. Estas redes podem ser classificadas em dois tipos [TELECO, 2006]:

- Públicas: a Internet representa a rede IP pública usada para comunicações VoIP. O usuário deve ter preferencialmente um acesso de banda larga (ADSL, cabo, rádio, Wimax, etc.) instalado para poder fazer uso do serviço VoIP.
- Privadas: correspondem as redes corporativas das empresas usadas para comunicações VoIP. Podem ser desde pequenas redes locais (*Local Area Network - LAN*) até grandes redes corporativas (*Wide Area Network - WAN*) de empresas com presença global.

A telefonia IP abrange o uso da aplicação VoIP para estabelecer chamadas telefônicas com a rede de telefonia pública (fixa ou celular). Os serviços de Telefonia IP existentes são diferenciados através das chamadas. Em um dos serviços é possível estabelecer chamadas para rede pública, onde o usuário discar o número convencional do telefone de destino para completar a chamada. No outro, é possível fazer e receber chamadas da rede pública. Neste caso o usuário recebe um número convencional de telefone, para receber as chamadas da rede pública, e discar o número convencional do telefone de destino para fazer a chamada para a rede pública. As Figuras 3.4 [TELECO, 2006] e 3.5 [TELECO, 2006] apresentam os dois casos de serviços.

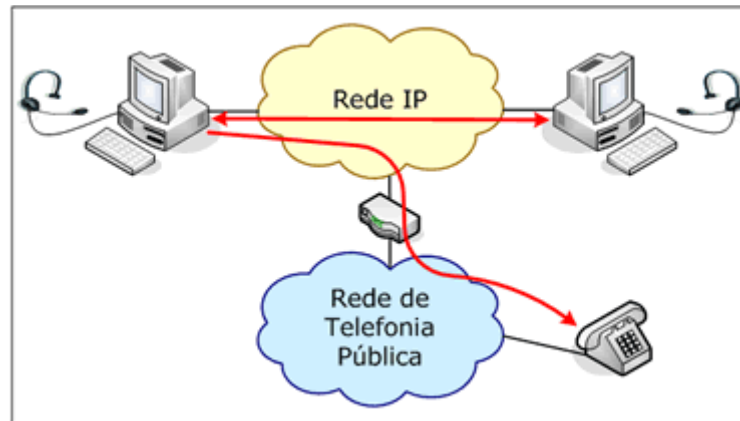


Figura 3.4 [TELECO, 2006] – realiza chamadas para a rede pública

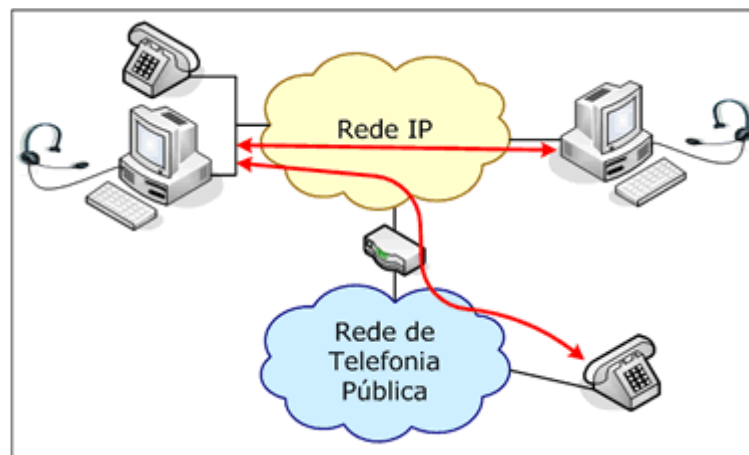


Figura 3.5 [TELECO, 2006] - realiza e recebe chamadas para rede pública.

Para ambos os casos, o usuário pode fazer e receber chamadas de outro usuário do mesmo provedor de serviços VoIP, geralmente sem custo, porém não consegue chamar usuários de outros provedores VoIP.

3.5. CONVERSÃO DE VOZ

Nos estudo sobre VoIP, deve-se estar atento a parte de codificação do sinal de voz para dados digitais, em virtude da voz ser um sinal analógico. Este processo de digitalização de um sinal de voz possibilita um modo mais eficiente no seu armazenamento, através do uso de um dispositivo denominado *codec*. Ele realiza a digitalização sinais de voz para transmissão por serviços de dados digitais e os converte novamente no outro ponto de transmissão.

Várias técnicas de foram geradas para codificação de voz. Cada técnica é representada por uma recomendação específica, padronizada pela ITU-T. Estas técnicas realizam a segmentação do sinal analógico em intervalos periódicos, para geração de quadros após a digitalização. Estes quadros são compostos por informações do sinal de voz deste período, além das informações de uma parcela do quadro subsequente. Um parâmetro importante neste processo é o *lookhead*, tempo necessário para coletar as informações do próximo quadro.

Inicialmente, foram criadas técnicas como a PCM, conhecidas como *codec* não linear e a ADPCM. Também foram geradas outras técnicas além das citadas anteriormente, como mostra a Tabela 3.2, que apresenta algumas características sobre cada técnica.

Tabela 3.2 - Especificações dos padrões de codificação do sinal de voz PCM e ADPCM, além de LD-CELP, CS-ACELP, MP-MLQ e ACELP.

Técnica de Codificação	Recomendação	Taxa (Kbps)	Quadro por lookahead (ms)
PCM	G.711	64	0,125 / 0
ADPCM	G.726	40, 32, 24, 16	0,125 / 0
LD-CELP	G.728	16	0,625 / 0
CS-ACELP	G.729	8	10 / 5
MP-MLQ	G.723.1	6,3	30 / 7,5
ACELP	G.723.1	5,3	30 / 7,5

Por ser uma tecnologia em desenvolvimento, o VoIP está sendo padronizado por diversas organizações. Uma delas, a *International Multimedia Teleconferencing Consortium* (IMTC) criou um fórum que discute sobre VoIP, com o objetivo de oferecer soluções de interoperabilidade para diferentes fornecedores [CHOWDHURY, 2002].

3.6. ARQUITETURA VOIP

A arquitetura do VoIP é dividida em quatro níveis, de acordo com a definição de organizações, como a ITU-T, a IMTC, além da *Internet Engineering Task Force* (IETF) e seus respectivos padrões. A Figura 3.6 mostra cada nível e seus respectivos padrões:

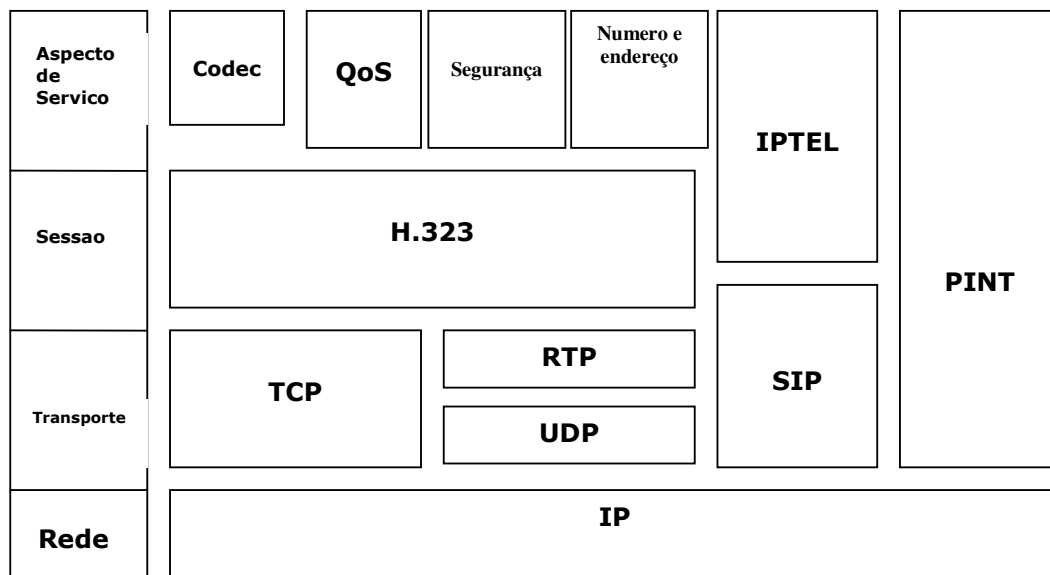


Figura 3.6 – A arquitetura em níveis do VoIP.

- **Nível de aspecto de serviço:** possui o papel de agregar todos os aspectos do serviço VoIP, como a segurança da cobrança e a codificação da fala em pacotes digitais.
- **Nível de sessão:** possibilita que o VoIP estabeleça uma chamada e realize um registro quando o terminal é conectado inicialmente rede.
- **Nível de transporte:** responsável pela remessa de mensagens de ponta a ponta.
- **Nível de rede:** neste nível ocorre a execução dos serviços de roteamento, como a transferência do pacote IP.

Cada atributo pertencente à arquitetura em níveis do VoIP foi definido por essas organizações. O protocolo de Rede de Telefonia Comutada Publica (*PSTN-Internet Interworking - PINT*) tem como função solicitar serviços telefônicos (fazer chamadas, enviar e receber fax e receber conteúdo pelo telefone) a partir de uma rede IP, com o objetivo de permitir que um cliente IP realize uma transação de chamada pela rede IP.

3.7. PROTOCOLOS DE COMUNICAÇÃO VOIP

A ITU-T criou uma recomendação que especifica terminais e outros elementos que oferecem sistemas de comunicação multimídia baseados em pacotes. Tal recomendação, denominada H.323, apresenta elementos que oferecem suporte para áudio, vídeo e dados, embora o áudio seja obrigatório, o vídeo e os dados são opcionais [LEOPOLDINO et al, 2007]. Esta recomendação focaliza controle de chamada, gerenciamento de multimídia e gerenciamento de largura de banda, além de interfaces entre LANs e outras redes. A rede através da qual os elementos H.323 se comunicam pode ser *unicast*, *multicast* ou um segmento múltiplo com topologias complexas. H.323 define padrões de multimídia para a infra-estrutura (ou seja, redes baseadas em IP) existente.

A recomendação H.323 aborda os requisitos técnicos para serviços de comunicação de áudio e vídeo em LANs que não oferecem suporte a QoS. Isso pode ser considerada uma das principais desvantagens desta recomendação, visto que a garantia de QoS é fundamental para o bom aproveitamento da totalidade dos recursos oferecidos pelas aplicações multimídias.

O Protocolo de Iniciação de Sessão (SIP), projetado pela IETF, é um protocolo de sinalização responsável por criar, modificar e terminar sessões com um ou mais participantes, incluindo distribuição de multimídia e conferência de Internet. Este protocolo é independente de protocolo de rede, além de ser orientado para Web, o que permite que ele aceite infra-estrutura Web, como segurança, *cookies* etc.

O SIP permite a mobilidade de usuário, através de Proxy e redirecionamento de requisições para o local atual do usuário. Os usuários desse protocolo são capazes de se comunicar uns com os outros através de *multicast* e *unicast*, ou uma combinação de ambos [CHOWDHURY, 2002].

Ele também não descreve como uma configuração deve ser gerenciada, em vez disso, ele usa um servidor central para gerenciar o estado da conferência e do participante, distribuindo esse estado via *multicast*. Ele pode convidar usuários para conferências, transportando as informações necessárias. Tal convite pode ser feito por protocolos *multicast*, por exemplo, o Protocolo de Divulgação de Serviço (*Service Advertisement Protocol – SAP*), e-mail, grupo de notícias, páginas Web ou catálogos, como o protocolo de atualização e pesquisa de diretórios (*Lightweight Directory Access Protocol – LDAP*). Deve-se notar que o SIP não aloca endereços *multicast*, isto é feito por um protocolo *multicast*, como o SAP.

O H.323 foi um importante protocolo para telefonia IP, mas cada vez mais vem perdendo espaço para o SIP. Os dois protocolos possuem vantagens e desvantagens, porém atualmente o SIP vem ganhando mais espaço na telefonia IP. Algumas das razões são:

- O protocolo H.323 é um protocolo robusto que foi inicialmente desenvolvido para aplicações multimídias em LAN's, diferente do SIP, que é um protocolo simples e eficiente, baseado nos protocolos: Protocolo de Transferência de Hipertexto (*HyperText Transfer Protocol - HTTP*) e Protocolo Simples para Transferência de Correio (*Simple Mail Transfer Protocol - SMTP*) da Internet.
- O H.323 possui uma complexidade muito maior que o SIP, uma vez que, utiliza diferentes protocolos e não é baseado em texto, é baseado em codificação binária ASN.1 PER. A maior complexidade do H.323 pode ser observada ao realizar uma chamada, o SIP envia apenas 4 pacotes, enquanto o H.323 precisa enviar 12 pacotes.
- O H.323 possui baixa integração com outros componentes da Internet e não oferece suporte a *firewall* nem a mensageiro instantâneo (*instant messenger*), pois, não foi inicialmente desenvolvido tendo como foco a Internet.
- Concluindo, o H.323 é um padrão muito poderoso, porém complexo demais para ser utilizado para telefonia IP. Uma vez que a tecnologia VoIP visa uma redução dos custos, o H.323 torna-se uma solução mais custosa, pois exige um grande esforço de implementação, diferente do SIP que é um protocolo simples, confiável e desenvolvido tendo como foco a Internet, ideal para telefonia IP. O fator decisivo para o SIP substituir o H.323 não está na qualidade e sim na complexidade.

Neste trabalho será focado o uso do protocolo SIP, devido suas vantagens e benefícios, expostos acima, para as aplicações VoIP, principalmente em cenários *Wi-Fi*, quando comparados com os dos H.323.

3.8. O PROTOCOLO RTP

O protocolo de transporte em tempo real (*Real-Time Transport Protocol – RTP*), definido pela RCF 3550 [CASNER et al, 2003], é um dos protocolos mais utilizados em VoIP. Em primeiro lugar, esse protocolo não oferece qualquer forma de controle de confiabilidade e congestionamento que existe no Protocolo de Controle de Transmissão

(*Transmission Control Protocol - TCP*). No entanto, RTP utiliza uma capacidade para incluir algum controle de fluxo, além de confiabilidade específica da aplicação. Embora o nome sugira que esse é um mecanismo de remessa em tempo real, ele não é um protocolo fim-a-fim. Deve-se notar que RTP por si só não oferece qualquer tipo de QoS ou remessa oportuna de dados. Sempre é necessário o suporte de um protocolo de nível inferior, que realmente tenha controle sobre recursos de *switches* e roteadores.

O RTP como mostra a Figura 4.3, trabalha em cima do Protocolo de Datagrama do Usuário (*User Datagram Protocol - UDP*). Isso pode ser conflitante, pois em algumas áreas, TCP pode ser mais apropriado do que UDP para o uso com o RTP. Entretanto, existem diversos motivos pelos quais TCP são impróprios para RTP:

- Durante a transmissão, alguns pacotes podem ser perdidos. Quando o receptor descobre os pacotes que faltam e os retransmite, os pacotes farão o caminho em direção ao transmissor e novamente para o receptor, ou então, será alocado um período maior para a transmissão. Portanto, o receptor precisa esperar pela retransmissão ou derrotar o mecanismo TCP descartando os pacotes retransmitidos. Em geral, TCP forçará a aplicação receptora a esperar. Esse comportamento do TCP induzirá um retardo maior para cada pacote na rede e, no fim, esse retardo seria drástico.
- O TCP não aceita *multicast*.
- O mecanismo de controle de congestionamento do TCP utiliza um procedimento de partida lenta, o que significa que isso reduz uma janela de congestionamento quando uma perda de pacotes for detectada. Por outro lado, não é possível diminuir taxas de transmissão de áudio e vídeo sem prejudicar o receptor.
- O cabeçalho TCP é muito maior do que o do UDP. Para transmissão de áudio em uma rede LAN, o TCP não oferece muita vantagem, pois a LAN oferece largura de banda suficiente e perda mínima de pacote.

Além de que apesar de ser um protocolo de transporte confiável, o TCP não suporta transmissão de voz em tempo real porque ele recupera os dados perdidos por retransmissão, assim o fornecimento dos dados, deve esperar por todas as retransmissões, gerando grandes atrasos.

Já o UDP não tem esse problema, pois fornece um serviço orientado a datagrama, mas tem a desvantagem de ser não-confiável. Como atraso tem que ser evitado em aplicações de voz

e na prática, a vazão máxima do 802.11b que uma aplicação VoIP pode obter é aproximadamente 5.9 Mbps utilizando TCP e 7.1 Mbps em UDP [ATHEROS, 2003], pode-se concluir que o UDP é a melhor escolha dentre os protocolos de transporte.

O RTP usa um protocolo de controle (*Real-Time Transport Control Protocol - RTCP*) para monitorar QoS e transportar informações em uma sessão contínua. Em geral, RTP possui duas partes: uma referente aos dados e outra é à funcionalidade de controle.

4. MÉTRICAS DE QUALIDADE PARA VOIP SOBRE REDES MESH

Para este estudo, foram utilizadas métricas as quais o VoIP apresenta maior sensibilidade para seu funcionamento. Essas métricas são: largura de banda, perda de pacotes, taxa de *jitter* e retardo.

4.1. QOS EM VOIP

Antes de tratar sobre a questão de Qualidade de Serviço (*Quality of Service - QoS*) em VoIP, é necessário também tratarmos de alguns conceitos fundamentais sobre Qualidade de Serviço. De acordo com a Organização Internacional de Padronização (*International Organization for Standardization - ISO*), a QoS de uma rede de comunicações pode ser conceituada como a qualidade pelo quanto a rede é capaz de atender as expectativas dos seus usuários através de seus serviços oferecidos. Ampliando tal conceito, a Qualidade de Serviço pode ser tratada como um modelo que envolve diversos aspectos relacionados à própria percepção do usuário (*Perceptual QoS*), aos requisitos das aplicações (*Application QoS*) e aos recursos disponíveis no sistema (*System QoS*), seja no equipamento do usuário (*Device QoS*) ou na própria rede (*Network QoS*) [NAHRSTEDT, 1995].

QoS pode ser obtida através de diversas técnicas de provisão. Entre elas, destaca-se o superdimensionamento da rede, que ofereceria de forma simples as garantias de QoS, porém exigiria uma infra-estrutura de alto custo para sua implantação. Outra forma seria a introdução de mecanismos que buscam racionalizar os recursos disponíveis na rede, onde tais mecanismos estabelecem diferentes níveis de serviço, cada um sendo tratado de acordo com sua prioridade estabelecida pelos seus requisitos de aplicação.

Esses requisitos das aplicações correspondem as suas necessidades para o seu bom funcionamento. Eles são traduzidos como parâmetros indicadores do desempenho da rede. Os parâmetros [TANENBAUM, 2003] podem ser:

- Confiabilidade: Todos os bits devem ser entregues de forma correta.
- Retardo (*delay*): período uniforme de atraso entre os pacotes enviados. Basicamente, seria o tempo consumido na transmissão de uma mensagem.
- Largura de Banda (*bandwidth*): quantidade de dados que podem ser transmitidos por unidade de tempo (*bps* – bits por segundo).

- *Vazão (throughput)*: quantidade de dados transmitida com sucesso por unidade de tempo (*bps* – bits por segundo).
- *Flutuação ou variação do atraso*: período irregular de atraso entre os pacotes enviados, denominado *jitter*.
- *Taxa de Erros ou de perda de pacotes (packet error/loss rate – PLR/PER)*: medida que define a razão entre a quantidade de pacotes perdidos e a quantidade de pacotes enviados.

A Tabela 4.3 apresenta os requisitos mínimos de QoS para cada aplicação:

Tabela 4.3 - [TANENBAUM, 2003] - Aplicações e seus requisitos de QoS.

Aplicação	Confiabilidade	Retardo	Flutuação	Larg. de Banda
Correio eletrônico	Alta	Baixa	Baixa	Baixa
Transf. de Arquivos	Alta	Baixa	Baixa	Média
Acesso a Web	Alta	Média	Baixa	Média
Login remoto	Alta	Média	Média	Baixa
Áudio por demanda	Baixa	Baixa	Alta	Média
Video por demanda	Baixa	Baixa	Alta	Alta
Telefonia	Baixa	Alta	Alta	Baixa
Videoconferência	Baixa	Alta	Alta	Alta

Os requisitos relativos à QoS variam de acordo com a natureza da aplicação. As aplicações de áudio e vídeo são sensíveis à vazão, ao retardo e à flutuação, porém podem admitir erros ou perdas de pacotes. Em aplicações de transferência de arquivos, os erros e perdas de pacotes podem causar sérios danos em relação à totalidade da mensagem transmitida, entretanto, podem aceitar um nível tolerante de retardo ou vazão. Ainda assim, é fundamental ter um bom entendimento da natureza e requisitos do tráfego multimídia para se obter um bom nível de QoS em redes com este tipo de tráfego. Este trabalho utilizará a aplicação VoIP para suas análises devido a sua sensibilidade com alguns aspectos (requisitos) da rede sem fio com o intuito de realizar um teste mais consistente sobre a qualidade de serviço da rede.

Baseado nestes parâmetros de QoS de cada aplicação, foram propostas classes de tráfego. Cada classe diferencia uma da outra de acordo com os requisitos e necessidades de cada aplicação. Alguns trabalhos [CHOI, 1999] tratam desta classificação de tráfegos como:

- Classe I: Suporte para aplicações de tempo real, onde se encontram aplicações multimídia, caracterizadas por serem sensíveis ao retardo, porém tolerantes a taxa de perda.
- Classe II-A: Também é sensível ao retardo, no entanto não necessita de garantia do mesmo. Nela encaixam-se aplicações com o *remote login*, *Secure Shell* (SSH), telnet etc.
- Classe II-B: Sensível a erros, porém não há restrições ao retardo. Aplicações comuns neste tipo de classe são as de transmissão de dados e de *e-mail*.

4.2. BANDA

Como foi visto anteriormente, a largura de banda é quantidade de dados que podem ser transmitidos por unidade de tempo, normalmente expressa em bits por segundo (bits/s, bps) ou em múltiplos dessa unidade (Kilobits/s, Megabits/s).

Em VoIP, a reserva de largura de banda é essencial para uma boa qualidade. A largura de banda estima a taxa de pacotes VoIP em função do tráfego. Os pacotes de voz experimentam em cada roteador, ou *gateway*, da rede um atraso aleatório resultado do processamento, da competição com outros serviços e do tratamento e esquema de prioridade das filas de pacotes [SOUZA, 2006]. Outro fator, é que um grande número de chamadas no tráfego da rede necessita de uma banda maior para suportar uma qualidade aceitável de voz.

Baseado nestes fatores, a rede deve oferecer uma maior largura de banda, de modo que possa atender a solicitação de várias chamadas VoIP. A utilização da banda varia significativamente através de rede WAN, logo, será necessária uma proposta mais complexa para estimar o uso da banda [PERCY et al 2003]. Algumas técnicas foram sugeridas para reduzir o uso de banda em sistemas VoIP. Elas utilizam a idéia de compressão do cabeçalho RTP (*compressed Real-time Transport Protocol – cRTP*), que comprime cabeçalhos utilizados pelo VoIP para reduzir a banda, além da Detecção de Atividade de Voz (*Voice Activity Detection – VAD*), que realiza o tratamento de transmissão de pacotes de voz vazio.

4.3. PERDA DE PACOTES

VoIP é excepcionalmente intolerante a perda de pacotes [KUHN, 2004] . A perda de pacotes pode resultar do excesso de latência, onde um grupo de pacotes chega atrasado e deve ser descartado para a chegada de pacotes mais novos. A perda também pode ser o resultado da variação de atraso, isto é, quando um pacote chega depois que o grupo a que ele pertencia já foi retirado do *buffer* receptor, tornando o pacote recebido inútil. As questões específicas de perda de pacotes existem além daquelas já associadas com redes de dados. Juntamente com problema de perda de pacote está a confiança do VoIP no RTP, que usa o UDP para o transporte, e assim não garante a entrega de pacote [BRATTLI, 1999]. Geralmente, a perda de um pacote VoIP pode acarretar uma maior probabilidade de perda dos pacotes restantes. Ou seja, pacotes VoIP não são perdidos isoladamente.

Os pacotes VoIP são muito pequenos, contendo geralmente uma variação entre 10 a 50 bytes, que é aproximadamente 12.5 - 62.5 ms [KUHN, 2004], com a maior parte de implementações tendendo a limites mais curtos.

A perda tolerável de pacotes VoIP não deve ultrapassar o limite de 3% de pacotes perdidos [CHUAH, 1998]. Isto significa que ao ultrapassar tal limite, a chamada pode não corresponder às necessidades dos usuários.

As causas para ocorrência de perda de pacotes podem ocorrer devido a fatores como:

- Falta de espaço no *buffer* – É quando um pacote chega e o *buffer* da estação já se encontra cheio, este pacote é descartado.
- Estouro do limite (*deadline*) de retardo – Para pacotes de voz existe um limite de retardo, e quando o mesmo é atingido o pacote é descartado, a perda é contabilizada e o sistema passa imediatamente para o próximo pacote.

Apesar da inviabilidade de usar um protocolo de entrega garantido como o TCP, existem algumas alternativas para o problema de perda de pacotes. Não se pode garantir que todos os pacotes serão entregues, mas se a banda estiver disponível, enviar informação redundante pode anular a possibilidade da perda. Tal banda não é sempre acessível e a informação redundante terá de ser processada, adicionando mais latência ao sistema e produzindo uma perda de pacote ainda maior. Perdas na rede e descartes no buffer de compensação de *jitter* não devem ultrapassar 1% para chamadas de alta qualidade

4.4. TAXA DE JITTER

Também denominada como variação de retardo, o *jitter* é a variação observada no conjunto de valores de retardo de unidades de dados consecutivas, isto é, o atraso de pacote não uniforme. Em sistemas VoIP, o *jitter* pode ser consequência da baixa largura de banda, o que afeta significativamente a provisão de QoS no sistema. A variação de atraso pode fazer com que os pacotes cheguem e sejam processados fora da seqüência. No Protocolo de Transporte em Tempo Real (RTP), utilizado em UDP, os pacotes fora de ordem não são reagrupados no nível de protocolo.

A variação máxima tolerável para aplicações VoIP deve girar em torno de 20 a 50 ms [LUSTOSA, 2005]. Quando a variação é alta, os pacotes chegam aos seus destinos rapidamente. A solução geral para controlar esta variação em sistemas VoIP é o uso de um *buffer*, mas este tem que liberar pacotes de voz pelo menos a cada 150 ms, logo as variações de atraso devem ser limitadas [HISERKORN, 2005]. A implementação de *buffers* é composta pela incerteza de não poder dizer se um pacote que está atrasado por um longo período de tempo, ou se está perdido. Isto significa que a taxa de *jitter* depende dos *buffers* adaptativos de compensação de *jitter* e dos outros atrasos que compõem o atraso fim-a-fim.

A variação também pode ser controlada em outras partes da rede VoIP através de roteadores, *firewalls*, e outros elementos de rede que oferecem QoS. Esses elementos processam e enviam tráfego urgente, como pacotes VoIP, mais rapidamente do que pacotes de dados menos urgentes. Outro método para reduzir variação de atraso é padronizar o tráfego da rede para diminuir a variação fazendo uso eficiente da banda. Este problema conflita com algumas medidas de segurança em VoIP. Dentre estes está o protocolo de segurança IP (*IP Security - IPSec*), cuja exigências de processamento podem aumentar a latência, assim limitando a eficácia da banda e contribuindo para a variação de atraso. A eficácia da banda é comprometida quando nos pacotes são adicionados novos cabeçalhos. No tráfego IP normal, este problema é despercebido desde que a mudança no tamanho do pacote seja muito pequena se comparada com o tamanho do pacote. Como VoIP usa pacotes muito pequenos, até um aumento mínimo é importante porque este aumento se acumula em todos os pacotes, e o serviço VoIP envia um volume muito alto desses pequenos pacotes.

A janela de entrega de um pacote VoIP é muito pequena, então ela assume que a variação aceitável no atraso do pacote é menor. Assegurar que atrasos na entrega de pacotes causados por dispositivos de segurança sejam mantidos uniforme em todo tráfego é importante. Implementação de dispositivos que suportam QoS e melhoram a eficiência da

banda com a compressão de cabeçalho permitem um atraso de pacote mais uniforme em uma rede VoIP segurada.

4.5. RETARDO

Pacotes de dados VoIP são extremamente sensíveis a retardo quando comparados a outras aplicações. O retardo é o fator principal para uma boa qualidade na transmissão de voz. É o que chamamos de período consumido para a transmissão de pacotes de voz.

A recomendação G.114 da ITU-T sugere que um “retardo de transmissão unidirecional” de até 400 ms é aceitável [CHOWDHURY, 2002]. Este padrão explica tal requisito de retardo da seguinte maneira:

- Um retardo de 0 a 150 ms é aceitável para a maioria das aplicações.
- Um retardo de 150 a 400 ms é aceitável desde que o projetista da rede esteja ciente do seu impacto em algumas aplicações do usuário.
- Acima de 400 ms é inaceitável.

O retardo fim-a-fim em VoIP é o resultado de alguns componentes de retardo, os principais são:

- Retardo de Propagação: causado pela distância e pelo meio dos enlaces envolvidos. Dentro de um país, o retardo de propagação unidirecional pelas linhas terrestres está abaixo de 18 ms.
- Retardo de Serialização: Esse é o tempo gasto para transmitir um pacote de voz de cada vez por um enlace de WAN. O retardo de serialização depende do tamanho do pacote de voz e da largura de banda do enlace.
- Retardo de Fila: Corresponde ao tempo gasto para um pacote esperar na fila de transmissão do enlace antes de ser serializado. Em um enlace onde os pacotes são processados na ordem de primeiro a chegar, primeiro a ser atendido, o tempo médio da fila em ms pode ser encontrado por:

$$\text{Atraso de serialização em ms} = 8 \times \frac{\text{tamanho do pacote IP em bytes}}{\text{largura de banda do link em kbps}}$$

O tamanho médio dos pacotes da *intranet* transportados por enlaces da WAN geralmente está entre 250 e 550 bytes. Os retardos na fila podem ser importantes para

enlaces com largura de banda abaixo de 512 kbps, enquanto, com enlaces de velocidade maior, eles podem permitir níveis de utilização maiores.

4.6. MOS

A medida de qualidade da voz transmitida é uma resposta subjetiva de um ouvinte. Uma medida comum usada para determinar a qualidade do som produzido pelos *CODECs* específicos é a da Pontuação de Opinião Média (*Mean Opinion Score* - MOS). Com o uso do MOS, um certo número de ouvintes julgam a qualidade de uma amostra de voz (correspondendo a um *CODEC* particular) numa escala de 1 a 5. A partir desses resultados, é calculada a média das pontuações (*scores*) para atribuir o MOS para aquela amostra [CISCO, 2006]. Na Tabela 4.4 são descritos as pontuações (*scores*) MOS. Já na Tabela 4.5 são apresentados alguns *scores* MOS para os *CODECs* mais usados.

Tabela 4.4 [DIGITAL AUDIO, 2007] - Pontuação MOS.

Pontuação	Definição	Descrição
5	Excelente	Um sinal de voz perfeito gravado em um local silencioso
4	Bom	Qualidade de uma chamada telefônica de longa distância (PSTN)
3	Razoável	Requer algum esforço na escuta
2	Pobre	Fala de baixa qualidade e difícil de entender
1	Ruim	Fala não clara, quebrada

Tabela 4.5 [CISCO, 2006] - Pontuação MOS de alguns CODEC's.

Método de Compressão	Bit Rate (kbit/s)	Pontuação MOS	Delay (ms)
G. 711 PCM	64	4.1	0.75
G.726 ADPCM	32	3.85	1

G.728 LD-CELP	16	3.61	3 para 5
G.729 CS-ACELP	8	3.92	10
G.729 x 2 Encodings	8	3.27	10
G.729 x 3 Encodings	8	2.68	10
G.729a CS-ACELP	8	3.7	10
G.723.1 MP-MLQ	6.3	3.9	30
G.723.1 ACELP	5.3	3.65	30

A medida MOS é uma medida subjetiva, portanto, seu resultado quanto a sua avaliação define uma fidelidade maior. Ainda assim, existem métodos objetivos que utilizam recursos computacionais para inferir VQ. Um dos principais métodos é o Modelo E [LUSTOSA, 2005]. Ele realiza uma análise dos fatores que influenciam a qualidade da voz, como atraso, perda, descarte no buffer de compensação de *jitter*, distorção inerente do *codec* etc. Cada um desses fatores é analisado separadamente. Ele ainda possui extensões que vêm sendo propostas para torná-lo mais fiel ao MOS [LUSTOSA, 2005]. A Figura 4.7 apresenta esta relação com o MOS:

R Satisfação do Usuário MOS

100	Muito Satisfeito	4.5
90	Satisfeito	4.3
80	Alguns usuários insatisfeitos	4.0
70	Muitos usuários insatisfeitos	3.6
60	Quase todos os usuários insatisfeitos	3.1
50	Não Recomendado	2.6
0		1

Figura 4.7[LUSTOSA, 2005] - Relação entre MOS e o Modelo E.

5. ESTUDO DE CASO

Neste capítulo é descrito o cenário que foi criado para a realização dos testes, os equipamentos e ferramentas utilizadas para a geração de chamadas VoIP sobre uma rede *mesh* e os resultados obtidos com os experimentos.

5.1. EQUIPAMENTOS

Para a criação da rede *mesh* utilizada no teste, foram usados quatro roteadores da linksys dos modelos WRT54GS versão quatro, apresentado na Figura 5.8, eles possuem as seguintes características:

- *Bootloader* : CFE version 1.0.37 for BCM947XX (32bit,SP,LE)
- *System-On-Chip* : Broadcom 5352EKP
- CPU: 200 Mhz
- *Mémoria Flash* : 4 MB (Intel TE28F320)
- *Mémoria RAM* : 16 MB (Hynix HY5DU281622ET)
- Placa de rede *Wi-Fi* : Integrated Broadcom BCM2050KML
- *Switch* : interno
- *boot_wait* : desligado



Figura 5.8 - Router WRT54GS v4.

Das características citadas cabe ainda ressaltar que estes equipamentos possuem os valores mínimos de memória que tornam possível a utilização satisfatória do OpenWRT. Outra observação deve ser feita a respeito da variável *boot_wait* que é criada em uma memória não volátil de acesso aleatório (*Non-Volatile Random Access Memory* - *NVRAM*). Diferentemente das versões anteriores, a versão quatro do WRT54GS, apresenta por padrão

esta variável com um valor falso. Tal variável deve ser alterada o mais breve possível dado sua extrema importância funcional. É esta variável que permite, no caso de alguma falha ao carregar um novo *firmware*, realizar a recuperação do mesmo utilizando algum cliente de TFTP¹².

5.2. FERRAMENTAS UTILIZADAS

Para implantação da rede *mesh* usada no teste foi necessário além da modificação do *firmware*, a adição do *olsrd* que é o serviço que torna possível a utilização do algoritmo de roteamento dinâmico, juntamente com ele foram instalados dois *plugins*, são eles:

olsrd-mod-httpinfo: implementação de um pequeno servidor de HTTP que apresenta uma serie de informações a respeito do *olsrd*.

olsrd-mod-dot-draw: gera uma saída no formato *dot*¹³ com dados referentes aos gráficos da topologia formada pelo *olsrd*.

Para a visualização em tempo real das figuras foi utilizado um *scripts perl*, chamado *olsr-topology-view.pl*¹⁴ que recebe os dados enviados pelo *olsrd-mod-dot-draw* e gera e exibe as imagens automaticamente.

Os testes com VoIP sobre a rede *mesh* fizeram uso do *Asterisk* que é um *software* livre que implementa em software os recursos encontrados em um agente de troca automático de ramais privados (*Private Automatic Branch Exchange - PABX*) utilizando VoIP. Cabe ressaltar que já existem pacotes com o *asterisk* compilado e pronto para a instalação via *ipkg*.

O *softphone* usado no lado cliente foi o *xten-lite*¹⁵, apresentado na Figura 5.9, por oferecer suporte a SIP e apresenta clientes disponíveis em múltiplas plataformas.

12 Protocolo de Transmissão de Arquivos Trivial

13 <http://www.graphviz.org/cgi-bin/man?dot>

14 <http://meshcube.org/nylon/utis/olsr-topology-view.pl>

15 <http://www.xten.com/index.php?menu=Products&smenu=xlite>



Figura 5.9 - Cliente SIP : Xten-lite

Para a simulação do tráfego de voz entre os nós da rede foi usado o *iperf*, que é uma ferramenta GPL, usada para análise do desempenho da rede, permitindo a verificação de banda, *jitter* e perda de pacotes [IPERF, 2007].

5.3. CENÁRIO

Neste estudo foram utilizados quatro roteadores, identificados por id1, id2, id3 e id4, dispostos em locais distintos, conforme a Figura 5.10. Cada roteador possui duas sub-redes, uma que realiza a interligação com a rede cabeada e outra que interliga os equipamentos sem fio. A Tabela 5.6 mostra as correlações entre IPs e roteadores.

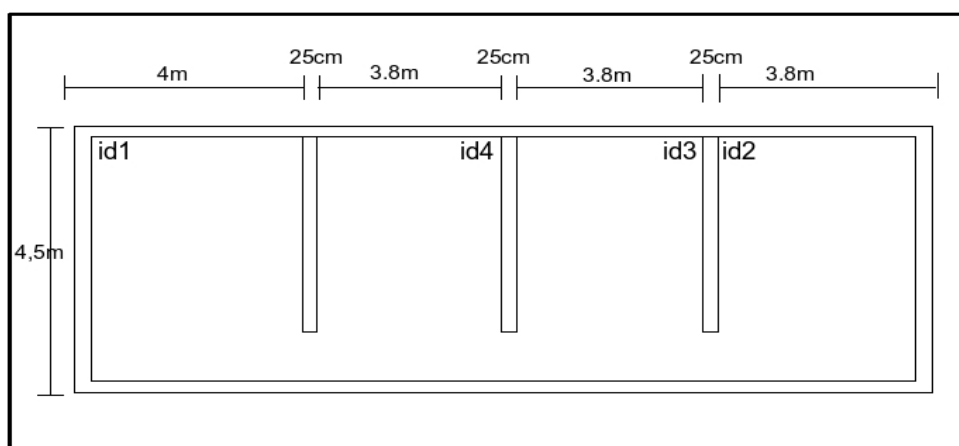


Figura 5.10 - Ambiente de teste

Tabela 5.6 - Relação IPs X Roteadores

Nome do roteador	IP <i>Wi-Fi</i>	IP <i>ethernet</i>
id1	10.151.1.1/255.255.0.0	10.152.0.33/255.255.255.224
id2	10.151.2.1/255.255.0.0	10.152.0.65/255.255.255.224
id3	10.151.3.1/255.255.0.0	10.152.0.97/255.255.255.224
id4	10.151.4.1/255.255.0.0	10.152.0.129/255.255.255.224

O esquema de endereçamento utilizado foi baseado nas propostas do professor [TSARMPOPOULOS et al., 2005] no projeto *VMesh*¹⁶. Este esquema de endereçamento supõe que cada nó seja um roteador e que a ele seja atribuído um identificador (ID). Desta forma, de acordo com o ID, são realizados cálculos que irão definir os endereços das interfaces de rede LAN e *Wireless*.

De acordo com os cálculos de endereçamento e as máscaras de rede utilizadas, tem-se a limitação impostas a cada nó de só pode ter 30 usuários conectados a ele, fornecendo aos mesmos endereços IP do seu intervalo via DHCP. De modo análogo, pode-se observar que é possível ter até 64516 nós interligados entre si, o que permite gerar uma rede mesh de tamanho consideravelmente grande [TSARMPOPOULOS et al., 2005].

O cenário inicial possuía a seguinte disposição lógica mostrada na Figura 5.11, obtida através do plugin *olsrd-mod-dot-draw*:

16 <http://vmesh.inf.uth.gr/>

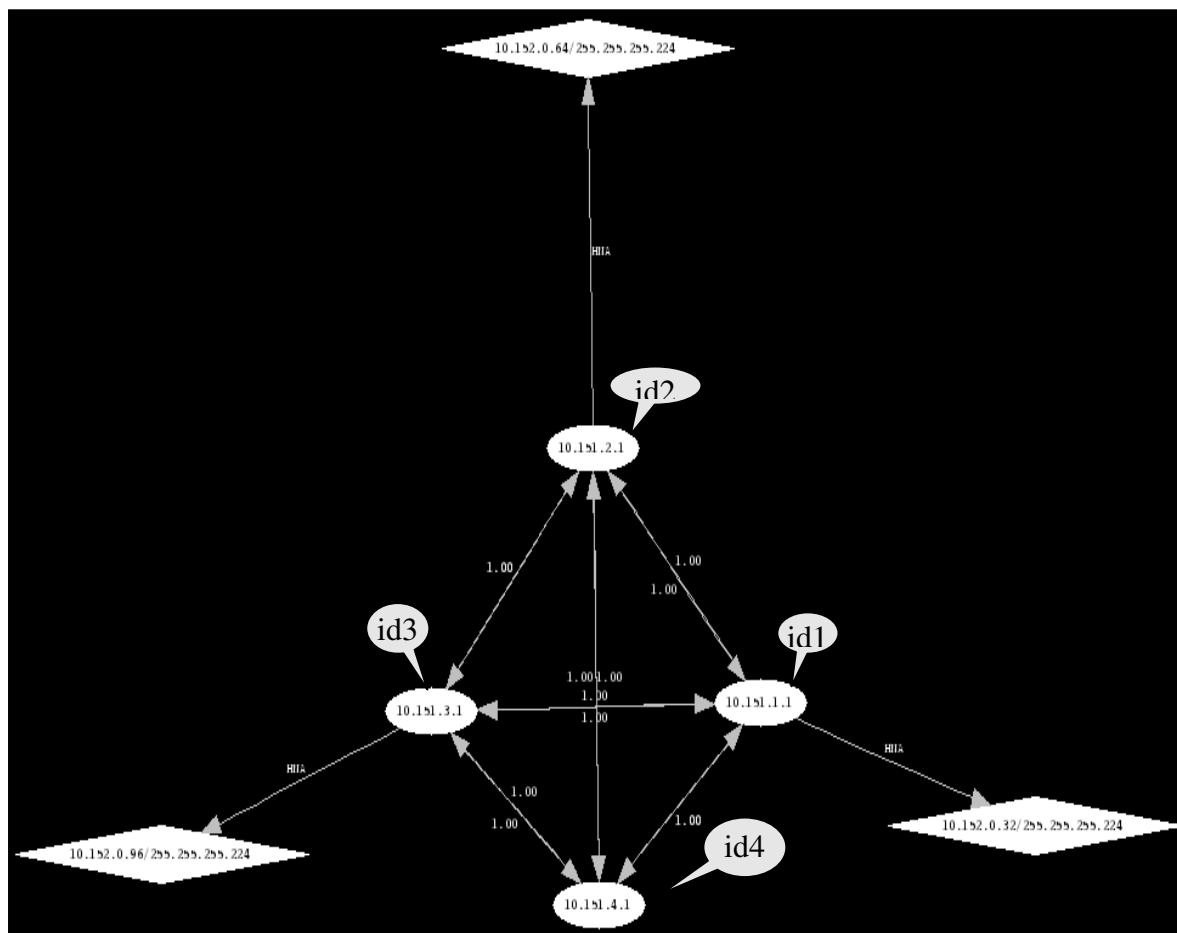


Figura 5.11 - Topologia com txpower igual a 19 db.

Este modelo é um grafo orientado e valorado possuindo como peso o valor do ETX de cada link representado pelas arestas, e tendo os retângulos e elipses representando os roteadores e os losangos representando as sub-redes de cada roteador.

Entretanto, esta topologia não pode ser usada nos teste, em virtude dos nós poderem se comunicar diretamente, sem a necessidade de múltiplos saltos, como um dos objetivos do teste é justamente verificar o impacto dos múltiplos saltos na qualidade de voz, foi necessário diminuir a potência do sinal de transmissão de 19 db para 1 db e inserir objetos, galões de água e caixas de papelão, que proporcionassem interferência para que a área de cobertura dos APs fosse menor. Com essas medidas foi possível obter o cenário que atenda os objetivos do trabalho.

A nova topologia obtida com a redução da potência do sinal de transmissão é mostrada na Figura 5.12, que simula uma disposição física em forma de “Y”:

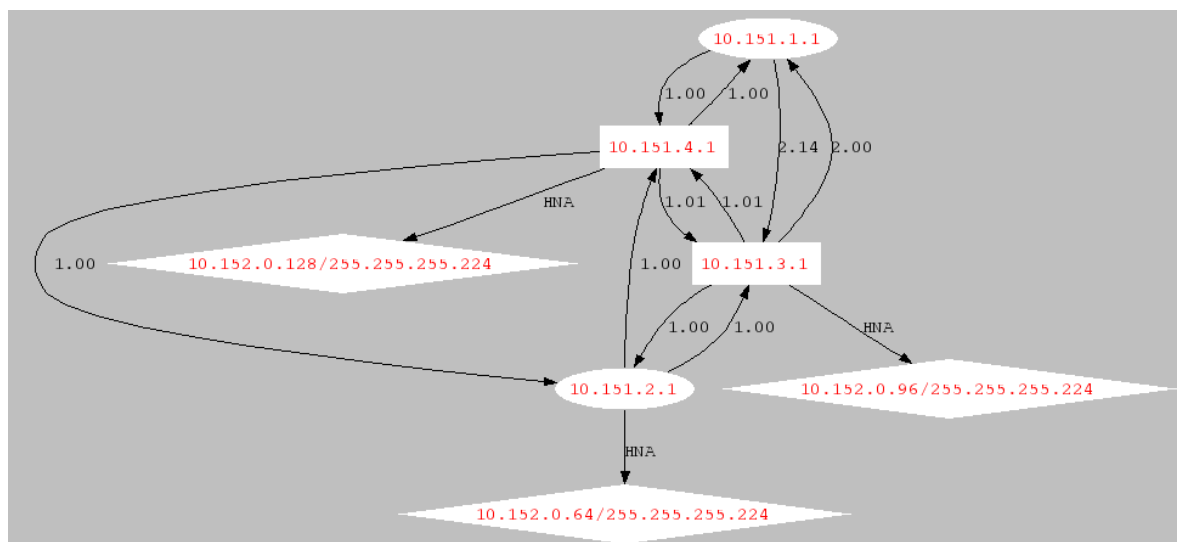


Figura 5.12 - Topologia lógica com 1db de potência.

5.4. EXPERIMENTOS

Para podermos avaliar o uso de VoIP me redes mesh dois experimentos foram realizados. No primeiro foi realizada uma chamada VoIP para testar a viabilidade e a qualidade de voz do serviço VoIP que a rede podia oferecer e no segundo foi feita a simulação de tráfego VoIP através da ferramenta *iperf* para verificar o número máximo de chamadas que a rede suporta.

5.4.1 Primeiro Experimento

Como primeiro teste sobre a rede *mesh*, foi realizada uma chamada VoIP entre dois clientes *xten-lite* cada um operando em máquinas distintas e conectadas uma no roteador id3 e outro no roteador id4. Para o teste foi utilizado um servidor *asterisk* instalado no roteador id2.

Esta configuração tem como principal objetivo testar a viabilidade do uso de VoIP, mesmo com a existência de múltiplos saltos, função esta exercida pelo roteador id4 e avaliar a qualidade da ligação de voz.

O *codec* GSM foi utilizado para realizar a compressão de voz no experimento devido seu relativamente baixo consumo de processamento e sua boa taxa de compreensão, o uso de um codec como o g729, apesar de sua excelente taxa de compreensão traria um consumo de processamento muito elevado que o dispositivo utilizado não possui.

5.4.2 Segundo Experimento

Para a simulação de chamadas voip foi utilizada a ferramenta *iperf*, que iria gerar o tráfego de acordo com parâmetros do *codec* G711. A escolha deste *codec* para a realização dos testes justifica-se pela baixa taxa de compressão deste que irá gerar um maior tráfego na rede. Este fato é interessante para verificar o comportamento da rede perante um tráfego intenso.

De acordo com [BENCHIMOL et al, 2005], o cálculo de banda consumida por uma ligação VoIP utilizando o *codec* de compressão de voz G.711 corresponde a 93.6 kbps. Este foi o valor usado nas simulações de tráfego.

A comunicação ocorreu entre os computadores conectados aos roteadores id1 e id3, tendo com isso um salto pelo id4. O *iperf* foi iniciado em modo servidor e esperando tráfego UDP no roteador id1 e em modo cliente com a opção de tráfego bidirecional no roteador id3, esta opção é necessária, pois a simulação de tráfego VoIP necessita possui dois fluxos de dados.

5.5. RESULTADOS

Neste trabalho foram focados os principais testes quanto às seguintes métricas: perda de pacotes (*packet loss*), variação de atraso (*jitter*) e MOS. De acordo com as análises realizadas, através das ferramentas de monitoração utilizadas na rede, foram obtidos resultados importantes para o este estudo, dado o cenário proposto.

Porém, para que se tenha uma compreensão maior destes resultados gerados, é necessário que esteja claro o entendimento dos seguintes tópicos:

- O cenário utilizado neste estudo possui limitações quanto ao número de roteadores, espaço e quanto à utilização de antenas externas para os testes *outdoor*.
- Não foram realizados teste com tráfego de *background*, sendo a rede dedicada ao tráfego VoIP.
- Como estamos trabalhando com redes sem fio existem muitos fatores que podem prejudicar os enlaces de comunicações como tráfego de pessoas e uso de aparelhos telefônicos sem fio.

5.6. RESULTADO DO EXPERIMENTO UM

Nas chamadas realizadas utilizando *asterisk* como servidor voip e os clientes *xten-lite* foi observada uma boa qualidade de transmissão de voz, sendo atribuída as ligações observadas um valor de MOS igual a 4. Podemos constatar com isso que o pouco processamento disponível nos roteadores (200MHz) não apresentou grande impacto sobre a compressão de voz usando o *codec* GSM. Cabe ressaltar que apenas uma chamada foi realizada e que para esta chamada a utilização tanto do *asterisk* sobre os dispositivos embarcados, quanto da rede *mesh* pra tráfego de voz foi viável.

5.7. RESULTADO DO EXPERIMENTO DOIS

Com a utilização do *iperf*, foram realizadas simulações de tráfego e testados os seguintes parâmetros: perda de pacotes (*packet loss*) e variação de atraso (*jitter*). Todos os gráficos foram propostos para diferentes números de chamadas. As chamadas para o cálculo do *jitter* ocorrem em um período de 0 a 10s com intervalos de amostragem de 0,5s, utilizando o protocolo UDP. As Figuras 5.13, 5.14, 5.15 e 5.16 apresentam os valores de *jitter* para

diferentes números de chamadas.

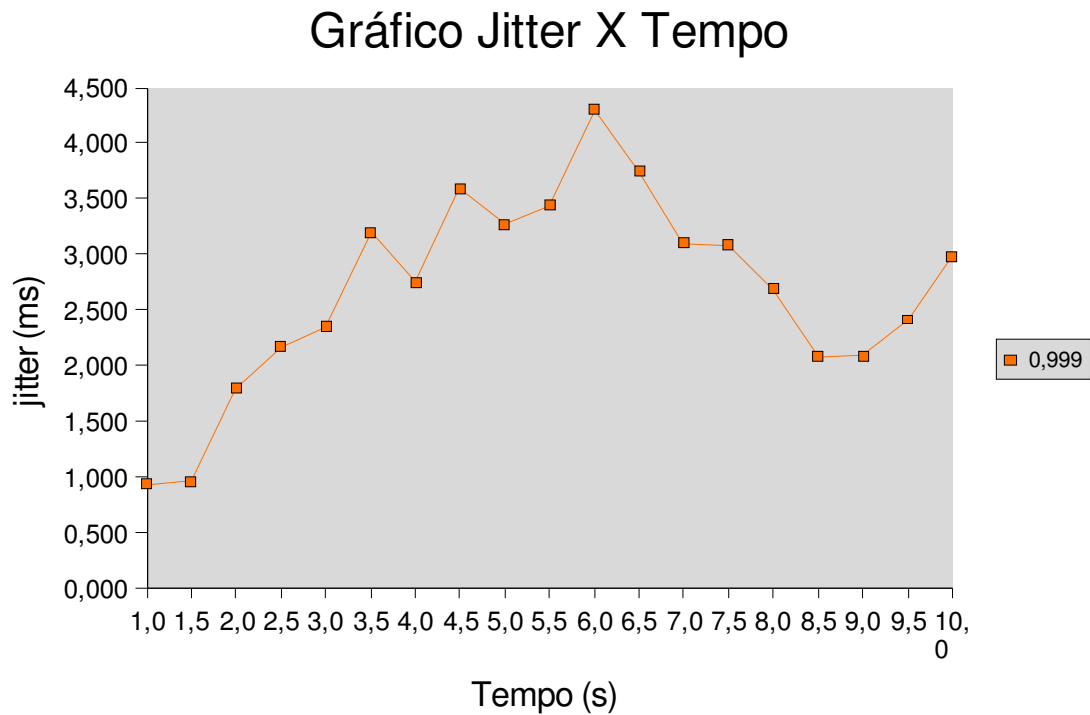


Figura 5.13 - Valores de jitter para uma chamada com duração de 10s utilizando o G.711.

Todas as medições são efetuadas no lado servidor, pois o mesmo apresenta um detalhamento destas informações, mostrando, também, o número de bytes recebidos. Enquanto que no lado do cliente é apenas informado um sumário no término da operação mostrando, também, quantos bytes o cliente tentou transferir.

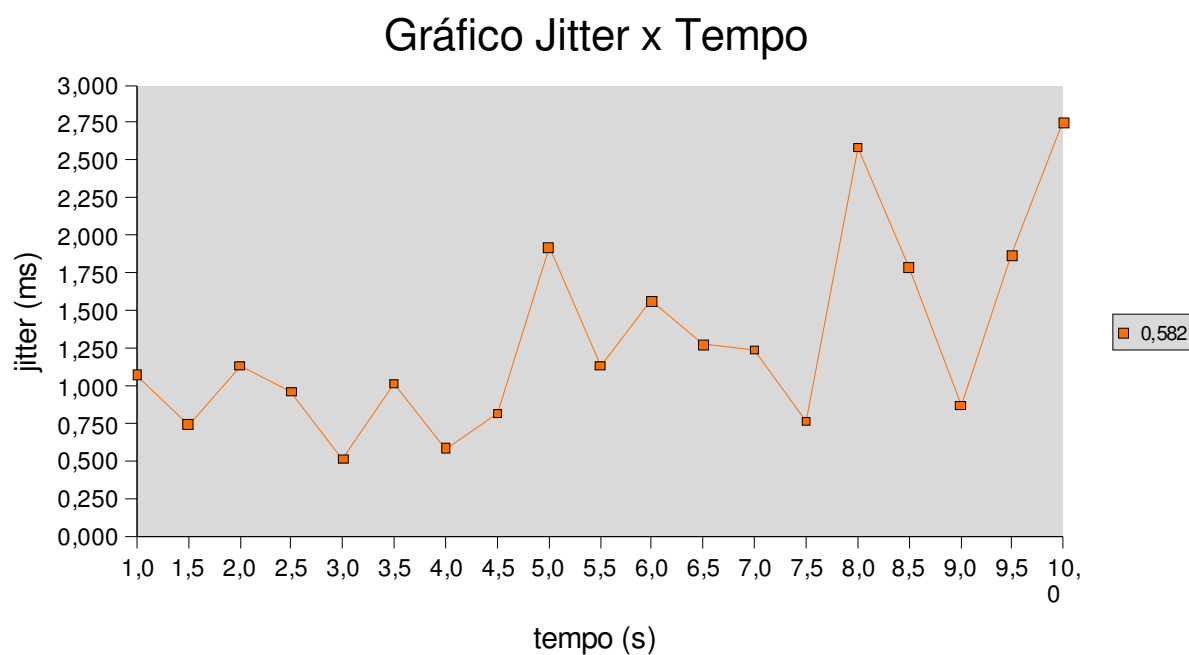


Figura 5.14 - Valores de jitter para dez chamadas com duração de 10s utilizando o G.711.

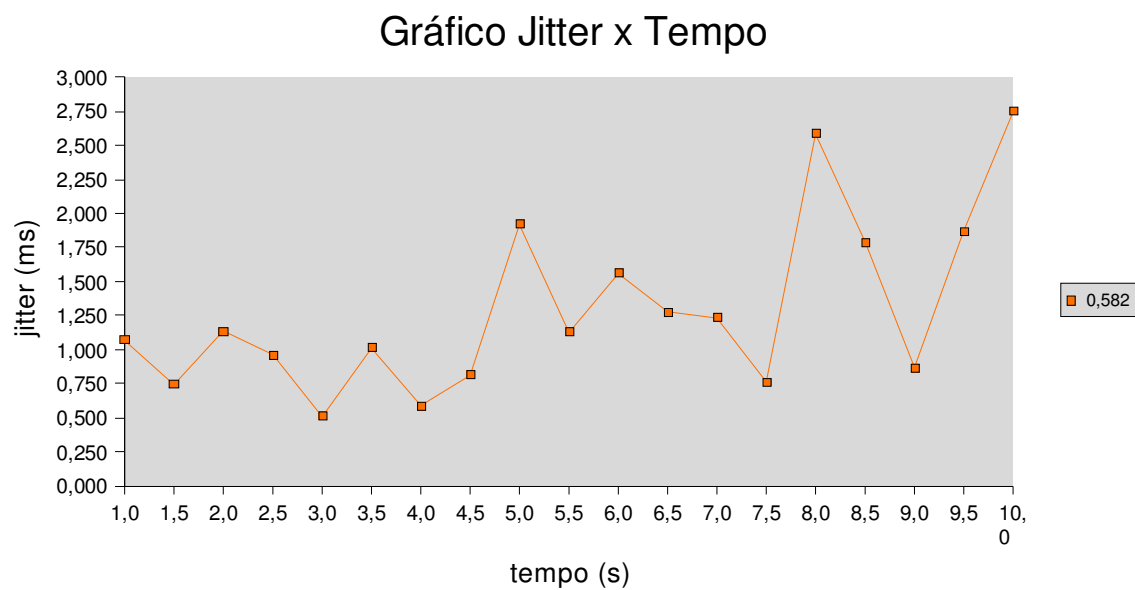


Figura 5.15 - Valores de jitter para vinte chamadas com duração de 10s utilizando o G.711.

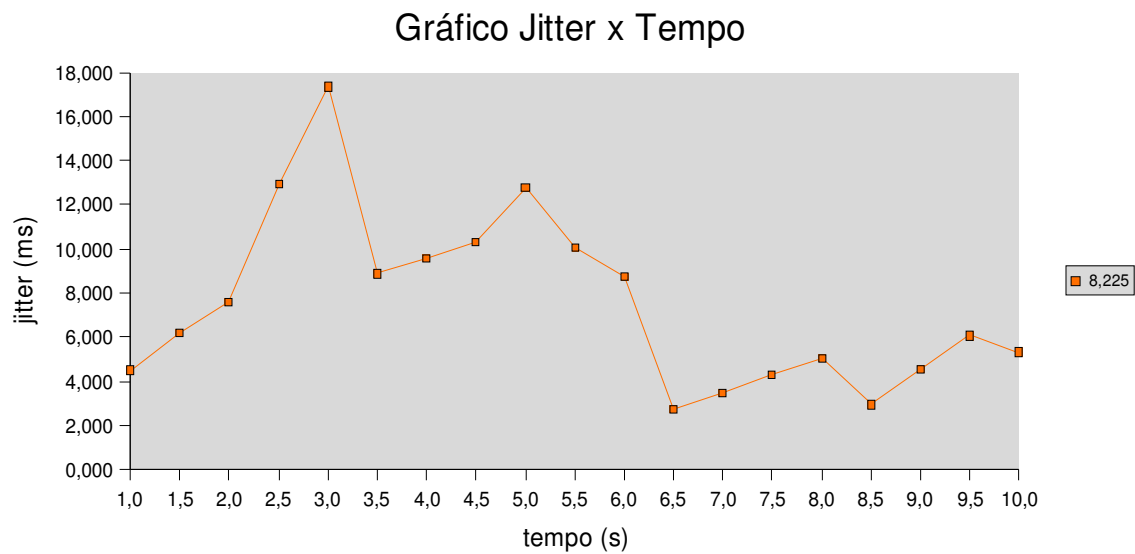


Figura 5.16 - Valores de jitter para quarenta chamadas com duração de 10s utilizando o G.711.

Através dos gráficos, verificamos que o *jitter* se encontra abaixo do limite máximo de 50 ms como afirmado por [LUSTOSA, 2005]. É possível constatar na Figura 5.15 que os valores de *jitter* foram inferiores ao da Figura 5.14, que representam, respectivamente, vinte e dez chamadas, tal variação devesse ao uso de redes sem fio, que podem apresentar interferência que degradam o sinal.

A perda também foi analisada através de um teste que inclui o número de chamadas em função da percentagem de perda. Esta análise é apresentada pela Figura 5.17.

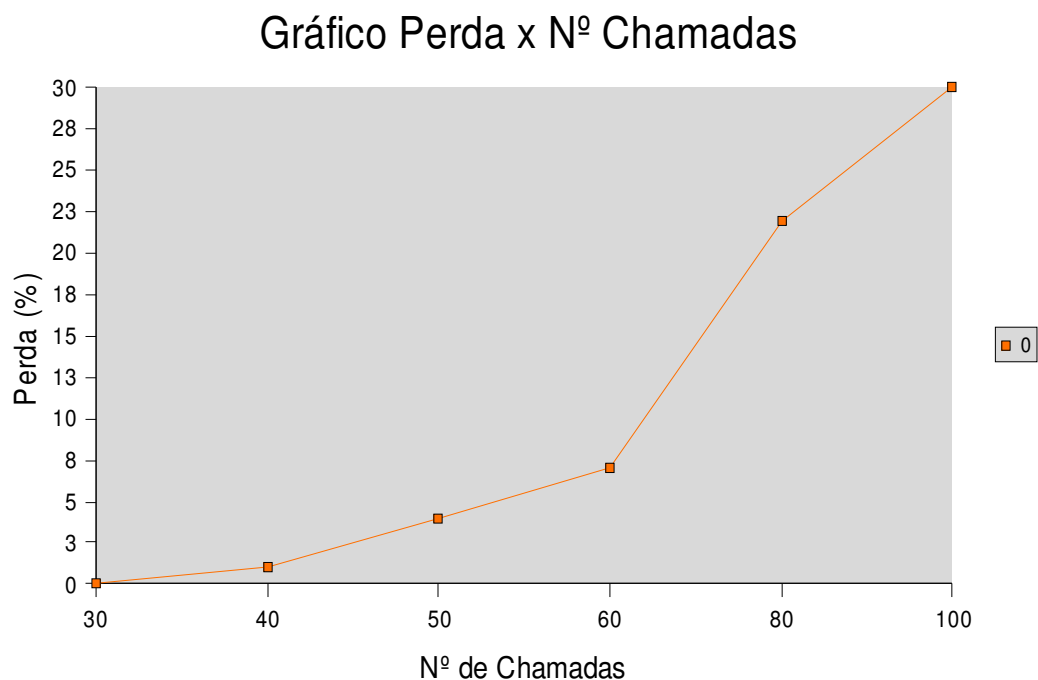


Figura 5.17 - Número de chamadas em função da perda para o G.711.

A análise de perda foi feita usando a razão entre os pacotes perdidos e os enviados. Portanto esta avaliação trabalha com uma media das chamadas realizadas.

Com a análise do Gráfico da Figura 5.17, é possível verificar que número de chamadas é proporcional à percentagem de perda de pacotes. Havendo assim um limite para o numero máximo de chamadas que a rede pode suporta.

Como a perda de pacotes não deve ultrapassar o limite de 3% [CHUAH, 1998], a análise do gráfico verifica que as chamadas são aceitáveis quanto a perda para até 40 chamadas VoIP, considerando que um pacote VoIP consome uma banda de 93,6 kbps [BENCHIMOL, 2005].

No entanto, quando é gerado um tráfego de igual ou acima de 50 chamadas percebe-se que a percentagem de perda de pacotes para este cenário ultrapassa o limite de 3%, chegando a 4% (Figura 5.17), o que afeta de forma significativa a qualidade da voz.

6. CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

6.1. CONSIDERAÇÕES FINAIS

Os objetivos propostos foram alcançados. Foi realizado um estudo teórico sobre redes em malha e constatamos seu contínuo avanço tanto na área acadêmica quanto no mercado. Os roteadores *Wi-fi* foram alterados e receberam novo sistema operacional e novo protocolo de roteamento, que caracterizam como roteadores *mesh*. Ocorreu a implantação de uma rede *mesh indoor*, com os equipamentos citados anteriormente, a qual foi utilizada para a realização de testes de VoIP.

Foram analisados os parâmetros (*jitter* e perda de pacotes) que mostraram a viabilidade do uso de até 40 ligações simultâneas dentro do cenário proposto. Contudo, ainda é necessário maiores testes, com o uso de tráfego de *background* e outros *codecs* para uma melhor avaliação do tráfego de voz sobre redes em malha.

Tais avaliações são muito importantes, pois esse tipo de rede vem conquistando um maior espaço como solução de acesso para a conectividade na ultima milha inclusive como uma proposta de infra-estrutura alternativa para a inclusão digital em zonas periféricas.

6.2. TRABALHOS FUTUROS

O caráter ainda evolutivo da tecnologia deixa um vasto campo de pesquisa a ser explorador e inúmeros problemas em aberto, como:

- O estudo a respeito de segurança em redes *mesh* sofre um serio agravante, pois além dos problemas de segurança já encontrados em redes sem fio tradicionais, não existe nem um mecanismo que garanta a segurança das informações que trafegam pelos nós da rede.
- Implementação de algoritmos de roteamento que viabilizem a prática de QoS. Existem poucos trabalhos na área e nem uma solução amplamente aceita.
- Estudos aprofundados a respeito do impacto de múltiplos saltos no desempenho da rede. Tal problema pode tornar inviável a criação de redes com muitos nós.
- Análise do comportamento do serviço de VoIP na presença de um tráfego de *background*.

REFERÊNCIAS

ABELÉM, Antônio J. G.; AGUIAR, Elisangela S.; BITTENCOURT, Paula; MOREIRA J., Waldir A. **Estudo comparativo de protocolos de roteamento para redes *Mesh* na região Amazônica – Relatório Técnico**, UFPA, 2006.

AKYILDIZ, Ian F.; WANG, Xudong; WANG, Weilin. **Wireless *mesh* networks: a survey**, 2005. Computer Networks. Disponível em <http://www.cs.ust.hk/~qianzh/COMP680H/reading-list/Akyildiz05.pdf>. Acessado em Janeiro de 2007.

ALBUQUERQUE de, Célio Vinícius Neves et al., **GT-*Mesh* - Relatório Técnico 1** - Termo de Referência e Estado da Arte, Rede Nacional de Pesquisas (RNP). Fevereiro 2006.

ALBUQUERQUE de, Célio Vinícius Neves et al., **GT-*Mesh* - Relatório Técnico 2 - Proposta do protótipo**, Rede Nacional de Pesquisas (RNP), Abril 2006. Wikipedia – **OpenWRT**. Disponível em <http://wiki.openwrt.org/OpenWrt>. Acessado em Janeiro de 2007.

ATHEROS COMMUNICATIONS. Methodology for Testing Wireless LAN Performance with Chariot. TechOnLine – Educational Resources, [S.l], out. 2003. Disponível em: http://www.super-ag.com/atheros_benchmark_whitepaper.pdf.>. Acessado em Janeiro de 2007.

BENCHIMOL, Bruno L. S.; MOREIRA J., Waldir A.; OLIVEIRA, Davis V. F. de. **Voz Sobre IP em Redes Sem Fio Padrão IEEE 802.11b**. 2005, 125f. Trabalho de Conclusão de Curso (Graduação em Bacharelado em Ciência da Computação), Universidade da Amazônia (UNAMA), Belém.

BRASIL, Robson. **Estudos de Tecnologia: Resumo de Tecnologias de acesso sem fio para uso em redes corporativas**, IPLANRIO/DIL/GTB, Agosto 2006. Disponível em

<http://www7.rio.rj.gov.br/cgm/ti/public_ipplan/publicacao/bit/arquivos/04.pdf>. Acessado em Janeiro de 2007.

BRATTLI, Thomas. Voice Quality (VQ) in Converging Telephony and Internet Protocol (IP) Networks. **International Engineering Consortium – On – Line Education**.1999. Disponível em <http://www.iec.org/online/tutorials/voice_qual/index.html>. Acessado em Janeiro de 2007.

BREUEL, Cristiano Malanga. **Redes em Malha sem Fios**. Instituto de Matemática e Estatística - Universidade de São Paulo (USP), Dezembro 2004. Disponível em: <http://grenoble.ime.usp.br/movel/Wireless_Mesh_Networks.pdf>. Acessado em Janeiro de 2007.

CASNER, S.; SCHULZRINNE, H.; FREDERICK, R.; JACOBSON, V. RTP: A Transport Protocol for Real-Time Applications. Request for Comments (RFC) 3550, 2003 Disponível em <<http://www.ietf.org/rfc/rfc3550.txt>>. Acessado em Janeiro de 2007.

CHOI, S. **QoS Guarantees in Wireless/Mobile Networks**. Tese de Doutorado, University of Michigan, 1999.

CHOWDHURY, Dhiman Deb. **Projetos Avançados de Redes IP**. trad.VIEIRA, Daniel. Editora Campus. Rio de Janeiro, 2002.

CHUAH, Chen-Nee; KATZ, Randy H. **Network Provisioning & Resource Management for IP Telephony**. **Electrical Engineering and Computer Science –Technical Reports**. University of California at Berkeley, 1998. Disponível em: <<http://www.eecs.berkeley.edu/Pubs/TechRpts/1999/CSD-99-1061.pdf>> Acessado em Janeiro de 2007.

CISCO Systems, **Understanding Codecs: Complexity, Hardware Support, MOS, and Negotiation**, 2006. Disponível em <http://www.cisco.com/en/US/tech/tk1077/technologies_tech_note09186a00800b6710.shtml>

[#mos](#)>. Acessado em Janeiro de 2007.

CLAUSEN T.; JACQUET, P. **Optimized Link State Routing Protocol (OLSR)** - RFC 3626, Outubro, 2003. Disponível em <http://ietf.org/rfc/rfc3626.txt>>. Acessado em Janeiro de 2007.

DIGITAL AUDIO SIGNAL PROCESSING LAB. **Online Mean Opinion Score Test**. Department of Electrical and Computer Engineering (EEN), University of Miami. Disponível em <http://umsis.miami.edu/~gliu/>>. Acessado em Janeiro de 2007.

HARADA, Eduardo. **Wireless Mesh Networks: Uma tecnologia que promete**, Disponível em <http://sisnema.com.br/Materias/idmat017459.htm>>. Acessado em Fevereiro de 2007.

HISERKORN, John. Timing is Everything for VoIP Success. **Electronic Engineering Times**. Janeiro de 2005. Disponível em http://www.eetasia.com/ARTICLES/2005JAN/B/2005JAN03_NTEK_TA.pdf>. Acessado em Janeiro de 2007.

IEEE - INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS. **Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications**. IEEE Standard 802.11, 1999.

IPERF, **Iperf**. The Board of Trustees of the University of Illinois. Disponível em <http://dast.nlanr.net/Projects/Iperf/>>. Acessado em Fevereiro de 2007.

IUT-T – International Telecommunication Union. **G.729 : Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear prediction (CS-ACELP)**. Disponível em <http://www.itu.int/rec/T-REC-G.729/e>>. Acessado em Fevereiro de 2007.

KUHN, Rick. **Voice Over Internet Protocol (VOIP) SECURITY**. National Institute of Standards and Technology, 2004. Disponível em http://csrc.nits.gov/ispab/2004-06/kuhn_2004_06_ispab.pdf>. Acessado em Janeiro de 2007.

KYASANUR, Pradeep; SO, Jungmin; CHEREDDI, Chandrakanth; VAIDYA Nitin H.; **Multi-Channel Mesh Networks: Challenges and Protocols**, University of Illinois at Urbana-Champaign. Disponível em <<http://www.hserus.net/~cck/pubs/wcom.pdf>>. Acessado em Janeiro de 2007.

LEOPOLDINO, Graciela Machado; MEDEIROS, Rosa Cristina Martins de. **H.323: Um padrão para sistemas de comunicação multimídia baseado em pacotes**. RNP – Rede Nacional de Ensino e Pesquisa. Disponível em <<http://www.rnp.br/newsgen/0111/h323.html>>. Acessado em Janeiro de 2007.

LUIZ, André; JÚNIOR, Olavo José Luiz. **Infra-estrutura e Roteamento em Redes Wireless Mesh**, Pontifícia Universidade Católica do Paraná – PUC-PR. 2005.

LUSTOSA, Leandro C. G. **Avaliação de Qualidade de Voz Em Sistemas baseados em VoIP**. Laboratório de Voz sobre IP - Universidade Federal do Rio de Janeiro (UFRJ). Disponível em <<http://www.voip.nce.ufrj.br/leandro/download/slides-Mestrado2005.pdf>>. Acessado em Janeiro de 2007.

NAHRSTEDT, Klara; STEINMETZ, Ralf. **Resource management in networked multimedia systems**. IEEE Computer. p.52-63, 1995. Disponível em <<http://ieeexplore.ieee.org/iel1/2/8715/00384118.pdf?tp=&isnumber=8715&arnumber=384118>>. Acessado em Janeiro de 2007.

OLSR DAEMON. Disponível em <<http://olsr.org/index.cgi?action=about>>. Acessado em Janeiro de 2007.

OPENWRT. **OpenWRT Wireless Freedom**. Disponível em <<http://openwrt.org/>>. Acessado em Outubro de 2006.

PERCY, Kenneth; HOMMER, Michael (Network Global Test Alliance). **Tips from the trenches on VoIP**. Network World. Janeiro de 2003. Disponível em

<<http://www.networkworld.com/research/2003/0127voip.html>>. Acessado em Janeiro de 2007.

PESSOA, Marcos de Lacerda; ROCHA, Marcelo Vinicius; FREITAS, Paulo Roberto de; JUSSIANI, Wagner Antonio. **Banda Larga sem Fio: Um Estudo para Curitiba**. Companhia Paranaense de Energia, Disponível em: <<http://www.copel.br/edicoes/4/004-01.pdf>>. Acessado em Janeiro de 2007.

RAMANATHAN, Ram; REDI, Jason. **A Brief Overview of Ad Hoc Networks: Challenges and Directions**, IEEE Communications Magazine, Maio 2002. Disponível em <http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1006968&isnumber=21724>. Acessado em Janeiro de 2007.

SOUZA, Jorge Moreira de. **QoS para VoIP I: Avaliação da Largura de Banda e do Atraso**. *Teleco* – Informação em telecomunicações. Dezembro de 2006. Disponível em <http://www.teleco.com.br/tutoriais/tutorialqosvoip1/pagina_1.asp>. Acessado em Janeiro de 2007.

TASK GROUP S. **STATUS of Project IEEE 802.11s**, Disponível em <http://www.ieee802.org/11/Reports/tgs_update.htm>. Acessado em Fevereiro de 2007.

TANENBAUM, Andrew. **Computer Networks**. 4.ed. United States:Prentice Hall, 2003. 912p.

TELECO - Informação e Telecomunicação. **Banda larga e VOIP**. Novembro de 2006. Disponível em <<http://www.teleco.com.br/voip.asp>>. Acessado em Janeiro de 2007.

TSARMPOPOULOS, N.; KALAVROS I. e LALIS S. **A Low-cost and Simple-to-Deploy Peer-to-Peer Wireless Network based on Open Source Linux Routers**, 2005. Disponível em <<http://ieeexplore.ieee.org/iel5/9524/30172/01386184.pdf?arnumber=1386184>>. Acessado em Fevereiro de 2007.

XUE, Qi; GANZ, Aura. **QoS routing for *Mesh*-based Wireless LANs**, Department of Electrical and Computer Engineering - University of Massachusetts. Disponível em <[http://www-unix.ecs.umass.edu/~qxue/publ/WMR_April\(IJWIN\).pdf](http://www-unix.ecs.umass.edu/~qxue/publ/WMR_April(IJWIN).pdf)>. Acessado em Fevereiro de 2007.

WIKI-OPENWRT, **OpenWrtDocs – OpenWRT Documentation**. Disponível em <<http://wiki.openwrt.org/>>. Acessado em Janeiro de 2007.

ANEXOS

Tempo (s)	Jitter (ms)
0,5	0,999
1,0	0,929
1,5	0,956
2,0	1,795
2,5	2,164
3,0	2,343
3,5	3,200
4,0	2,749
4,5	3,587
5,0	3,273
5,5	3,440
6,0	4,298
6,5	3,743
7,0	3,099
7,5	3,079
8,0	2,684
8,5	2,081
9,0	2,084
9,5	2,414
10,0	2,985

Valores de *jitter* para uma chamada de 10s de duração no G.711.

Tempo (s)	Jitter (ms)
0,5	5,165
1,0	3,505
1,5	5,941
2,0	4,413
2,5	4,310
3,0	4,281
3,5	3,990
4,0	7,399
4,5	9,750
5,0	5,987
5,5	4,547
6,0	4,697
6,5	5,085
7,0	5,610
7,5	6,300
8,0	5,686
8,5	9,848
9,0	10,265
9,5	10,252
10,0	10,110

Valores de *jitter* para dez chamadas de 10s de duração no G 711

Tempo (s)	Jitter (ms)
0,5	0,582
1,0	1,073
1,5	0,744
2,0	1,134
2,5	0,960
3,0	0,512
3,5	1,013
4,0	0,585
4,5	0,818
5,0	1,922
5,5	1,131
6,0	1,563
6,5	1,273
7,0	1,236
7,5	0,764
8,0	2,588
8,5	1,786
9,0	0,867
9,5	1,867
10,0	2,751

Valores de *jitter* para vinte chamadas de 10s de duração no

Tempo (s)	Jitter (ms)
0,5	8,225
1,0	4,500
1,5	6,211
2,0	7,626
2,5	12,940
3,0	17,357
3,5	8,873
4,0	9,567
4,5	10,297
5,0	12,782
5,5	10,042
6,0	8,780
6,5	2,751
7,0	3,482
7,5	4,269
8,0	5,015
8,5	2,944
9,0	4,548
9,5	6,074
10,0	5,324

Valores de *jitter* para quarenta chamadas de 10s de duração no

Número de chamadas	Percentagem de perda
20	0,00%
30	0,00%
40	1,00%
50	4,00%
60	7,00%
80	22,00%
100	30,00%

Valores de Perda em relação ao número de chamadas.

```

[ 5] 0.0- 0.5 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 0.0- 0.5 sec 5.74 KBytes 94.1 Kbits/sec 0,999 ms 0/ 4 (0%)
[ 5] 0.5- 1.0 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 0.5- 1.0 sec 4.31 KBytes 70.6 Kbits/sec 0,929 ms 0/ 3 (0%)
[ 5] 1.0- 1.5 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 1.0- 1.5 sec 5.74 KBytes 94.1 Kbits/sec 0,956 ms 0/ 4 (0%)
[ 5] 1.5- 2.0 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 1.5- 2.0 sec 7.18 KBytes 118 Kbits/sec 1,795 ms 0/ 5 (0%)
[ 4] 2.0- 2.5 sec 4.31 KBytes 70.6 Kbits/sec 2,164 ms 0/ 3 (0%)
[ 5] 2.0- 2.5 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 2.5- 3.0 sec 5.74 KBytes 94.1 Kbits/sec 2,343 ms 0/ 4 (0%)
[ 5] 2.5- 3.0 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 3.0- 3.5 sec 5.74 KBytes 94.1 Kbits/sec 3,200 ms 0/ 4 (0%)
[ 5] 3.0- 3.5 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 3.5- 4.0 sec 5.74 KBytes 94.1 Kbits/sec 2,749 ms 0/ 4 (0%)
[ 5] 3.5- 4.0 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 4.0- 4.5 sec 5.74 KBytes 94.1 Kbits/sec 3,587 ms 0/ 4 (0%)
[ 5] 4.0- 4.5 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 4.5- 5.0 sec 5.74 KBytes 94.1 Kbits/sec 3,273 ms 0/ 4 (0%)
[ 5] 4.5- 5.0 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 5.0- 5.5 sec 5.74 KBytes 94.1 Kbits/sec 3,440 ms 0/ 4 (0%)
[ 5] 5.0- 5.5 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 5.5- 6.0 sec 5.74 KBytes 94.1 Kbits/sec 4,298 ms 0/ 4 (0%)
[ 5] 5.5- 6.0 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 6.0- 6.5 sec 5.74 KBytes 94.1 Kbits/sec 3,743 ms 0/ 4 (0%)
[ 5] 6.0- 6.5 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 6.5- 7.0 sec 5.74 KBytes 94.1 Kbits/sec 3,099 ms 0/ 4 (0%)
[ 5] 6.5- 7.0 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 7.0- 7.5 sec 5.74 KBytes 94.1 Kbits/sec 3,079 ms 0/ 4 (0%)
[ 5] 7.0- 7.5 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 7.5- 8.0 sec 5.74 KBytes 94.1 Kbits/sec 2,684 ms 0/ 4 (0%)
[ 5] 7.5- 8.0 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 8.0- 8.5 sec 5.74 KBytes 94.1 Kbits/sec 2,081 ms 0/ 4 (0%)
[ 5] 8.0- 8.5 sec 5.74 KBytes 94.1 Kbits/sec
[ 5] 8.5- 9.0 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 8.5- 9.0 sec 5.74 KBytes 94.1 Kbits/sec 2,084 ms 0/ 4 (0%)
[ 4] 9.0- 9.5 sec 5.74 KBytes 94.1 Kbits/sec 2,414 ms 0/ 4 (0%)
[ 5] 9.0- 9.5 sec 5.74 KBytes 94.1 Kbits/sec
[ 4] 9.5-10.0 sec 5.74 KBytes 94.1 Kbits/sec 2,985 ms 0/ 4 (0%)
[ 5] 9.5-10.0 sec 5.74 KBytes 94.1 Kbits/sec
[ 5] 0.0-10.2 sec 116 KBytes 93.3 Kbits/sec
[ 5] Sent 81 datagrams
[ 4] 0.0-10.2 sec 116 KBytes 93.5 Kbits/sec 3.715 ms 0/ 81 (0%)
[ 5] Server Report:
[ 5] 0.0-10.2 sec 116 KBytes 93.4 Kbits/sec 1.416 ms 0/ 81 (0%)

```

Relatório do *Iperf* para uma chamada.

[5]	0.0- 0.5 sec	28.7 KBytes	470 Kbits/sec				
[3]	0.0- 0.5 sec	27.3 KBytes	447 Kbits/sec	2.642 ms	0/	19 (0%)	
[5]	0.5- 1.0 sec	28.7 KBytes	470 Kbits/sec				
[3]	0.5- 1.0 sec	33.0 KBytes	541 Kbits/sec	2.417 ms	0/	23 (0%)	
[5]	1.0- 1.5 sec	28.7 KBytes	470 Kbits/sec				
[3]	1.0- 1.5 sec	23.0 KBytes	376 Kbits/sec	2.235 ms	0/	16 (0%)	
[5]	1.5- 2.0 sec	28.7 KBytes	470 Kbits/sec				
[3]	1.5- 2.0 sec	34.5 KBytes	564 Kbits/sec	2.900 ms	0/	24 (0%)	
[5]	2.0- 2.5 sec	28.7 KBytes	470 Kbits/sec				
[3]	2.0- 2.5 sec	21.5 KBytes	353 Kbits/sec	2.342 ms	0/	15 (0%)	
[5]	2.5- 3.0 sec	28.7 KBytes	470 Kbits/sec				
[3]	2.5- 3.0 sec	34.5 KBytes	564 Kbits/sec	3.440 ms	0/	24 (0%)	
[5]	3.0- 3.5 sec	27.3 KBytes	447 Kbits/sec				
[3]	3.0- 3.5 sec	20.1 KBytes	329 Kbits/sec	7.685 ms	0/	14 (0%)	
[5]	3.5- 4.0 sec	28.7 KBytes	470 Kbits/sec				
[3]	3.5- 4.0 sec	37.3 KBytes	612 Kbits/sec	3.853 ms	0/	26 (0%)	
[5]	4.0- 4.5 sec	28.7 KBytes	470 Kbits/sec				
[3]	4.0- 4.5 sec	18.7 KBytes	306 Kbits/sec	7.792 ms	0/	13 (0%)	
[5]	4.5- 5.0 sec	28.7 KBytes	470 Kbits/sec				
[3]	4.5- 5.0 sec	37.3 KBytes	612 Kbits/sec	3.974 ms	0/	26 (0%)	
[5]	5.0- 5.5 sec	28.7 KBytes	470 Kbits/sec				
[3]	5.0- 5.5 sec	18.7 KBytes	306 Kbits/sec	8.418 ms	0/	13 (0%)	
[5]	5.5- 6.0 sec	28.7 KBytes	470 Kbits/sec				
[3]	5.5- 6.0 sec	37.3 KBytes	612 Kbits/sec	5.748 ms	0/	26 (0%)	
[5]	6.0- 6.5 sec	28.7 KBytes	470 Kbits/sec				
[3]	6.0- 6.5 sec	18.7 KBytes	306 Kbits/sec	4.167 ms	0/	13 (0%)	
[5]	6.5- 7.0 sec	28.7 KBytes	470 Kbits/sec				
[3]	6.5- 7.0 sec	37.3 KBytes	612 Kbits/sec	2.380 ms	0/	26 (0%)	
[5]	7.0- 7.5 sec	28.7 KBytes	470 Kbits/sec				
[3]	7.0- 7.5 sec	31.6 KBytes	517 Kbits/sec	1.823 ms	0/	22 (0%)	
[5]	7.5- 8.0 sec	28.7 KBytes	470 Kbits/sec				
[3]	7.5- 8.0 sec	24.4 KBytes	400 Kbits/sec	2.627 ms	0/	17 (0%)	
[5]	8.0- 8.5 sec	27.3 KBytes	447 Kbits/sec				
[3]	8.0- 8.5 sec	34.5 KBytes	564 Kbits/sec	2.353 ms	0/	24 (0%)	
[5]	8.5- 9.0 sec	28.7 KBytes	470 Kbits/sec				
[3]	8.5- 9.0 sec	23.0 KBytes	376 Kbits/sec	3.913 ms	0/	16 (0%)	
[5]	9.0- 9.5 sec	28.7 KBytes	470 Kbits/sec				
[3]	9.0- 9.5 sec	33.0 KBytes	541 Kbits/sec	2.397 ms	0/	23 (0%)	
[5]	9.5-10.0 sec	28.7 KBytes	470 Kbits/sec				

Relatório do *Iperf* para cinco chamadas.

[5]	0.0- 0.5 sec	57.4 KBytes	941 Kbits/sec				
[3]	0.0- 0.5 sec	53.1 KBytes	870 Kbits/sec	5,165 ms	0/	37 (0%)	
[5]	0.5- 1.0 sec	57.4 KBytes	941 Kbits/sec				
[3]	0.5- 1.0 sec	47.4 KBytes	776 Kbits/sec	3,505 ms	0/	33 (0%)	
[5]	1.0- 1.5 sec	57.4 KBytes	941 Kbits/sec				
[3]	1.0- 1.5 sec	64.6 KBytes	1.06 Mbits/sec	5,941 ms	0/	45 (0%)	
[5]	1.5- 2.0 sec	57.4 KBytes	941 Kbits/sec				
[3]	1.5- 2.0 sec	54.6 KBytes	894 Kbits/sec	4,413 ms	0/	38 (0%)	
[5]	2.0- 2.5 sec	56.0 KBytes	917 Kbits/sec				
[3]	2.0- 2.5 sec	58.9 KBytes	964 Kbits/sec	4,310 ms	0/	41 (0%)	
[5]	2.5- 3.0 sec	57.4 KBytes	941 Kbits/sec				
[3]	2.5- 3.0 sec	66.0 KBytes	1.08 Mbits/sec	4,281 ms	0/	46 (0%)	
[5]	3.0- 3.5 sec	57.4 KBytes	941 Kbits/sec				
[3]	3.0- 3.5 sec	45.9 KBytes	753 Kbits/sec	3,990 ms	0/	32 (0%)	
[5]	3.5- 4.0 sec	57.4 KBytes	941 Kbits/sec				
[3]	3.5- 4.0 sec	54.6 KBytes	894 Kbits/sec	7,399 ms	0/	38 (0%)	
[5]	4.0- 4.5 sec	57.4 KBytes	941 Kbits/sec				
[3]	4.0- 4.5 sec	58.9 KBytes	964 Kbits/sec	9,750 ms	0/	41 (0%)	
[5]	4.5- 5.0 sec	56.0 KBytes	917 Kbits/sec				
[3]	4.5- 5.0 sec	60.3 KBytes	988 Kbits/sec	5,987 ms	0/	42 (0%)	
[5]	5.0- 5.5 sec	57.4 KBytes	941 Kbits/sec				
[3]	5.0- 5.5 sec	51.7 KBytes	847 Kbits/sec	4,547 ms	0/	36 (0%)	
[5]	5.5- 6.0 sec	57.4 KBytes	941 Kbits/sec				
[3]	5.5- 6.0 sec	74.6 KBytes	1.22 Mbits/sec	4,697 ms	0/	52 (0%)	
[5]	6.0- 6.5 sec	57.4 KBytes	941 Kbits/sec				
[3]	6.0- 6.5 sec	37.3 KBytes	612 Kbits/sec	5,085 ms	0/	26 (0%)	
[5]	6.5- 7.0 sec	57.4 KBytes	941 Kbits/sec				
[3]	6.5- 7.0 sec	74.6 KBytes	1.22 Mbits/sec	5,610 ms	0/	52 (0%)	
[5]	7.0- 7.5 sec	56.0 KBytes	917 Kbits/sec				
[3]	7.0- 7.5 sec	37.3 KBytes	612 Kbits/sec	6,300 ms	0/	26 (0%)	
[5]	7.5- 8.0 sec	57.4 KBytes	941 Kbits/sec				
[3]	7.5- 8.0 sec	74.6 KBytes	1.22 Mbits/sec	5,686 ms	0/	52 (0%)	
[5]	8.0- 8.5 sec	57.4 KBytes	941 Kbits/sec				
[3]	8.0- 8.5 sec	38.8 KBytes	635 Kbits/sec	9,848 ms	0/	27 (0%)	
[5]	8.5- 9.0 sec	57.4 KBytes	941 Kbits/sec				
[3]	8.5- 9.0 sec	53.1 KBytes	870 Kbits/sec	10,265 ms	0/	37 (0%)	
[5]	9.0- 9.5 sec	57.4 KBytes	941 Kbits/sec				
[3]	9.0- 9.5 sec	58.9 KBytes	964 Kbits/sec	10,252 ms	0/	41 (0%)	
[5]	9.5-10.0 sec	56.0 KBytes	917 Kbits/sec				

Relatório do *Iperf* para dez chamadas.

[5]	0.0- 0.5 sec	86.1 KBytes	1.41 Mb/s				
[4]	0.0- 0.5 sec	67.5 KBytes	1.11 Mb/s	2.794 ms	0/	47 (0%)	
[5]	0.5- 1.0 sec	86.1 KBytes	1.41 Mb/s				
[4]	0.5- 1.0 sec	84.7 KBytes	1.39 Mb/s	3.018 ms	0/	59 (0%)	
[5]	1.0- 1.5 sec	84.7 KBytes	1.39 Mb/s				
[4]	1.0- 1.5 sec	106 KBytes	1.74 Mb/s	4.412 ms	0/	74 (0%)	
[5]	1.5- 2.0 sec	86.1 KBytes	1.41 Mb/s				
[4]	1.5- 2.0 sec	58.9 KBytes	964 Kbits/sec	2.986 ms	0/	41 (0%)	
[5]	2.0- 2.5 sec	86.1 KBytes	1.41 Mb/s				
[4]	2.0- 2.5 sec	109 KBytes	1.79 Mb/s	2.041 ms	0/	76 (0%)	
[5]	2.5- 3.0 sec	86.1 KBytes	1.41 Mb/s				
[4]	2.5- 3.0 sec	56.0 KBytes	917 Kbits/sec	5.287 ms	0/	39 (0%)	
[5]	3.0- 3.5 sec	84.7 KBytes	1.39 Mb/s				
[4]	3.0- 3.5 sec	109 KBytes	1.79 Mb/s	2.307 ms	0/	76 (0%)	
[5]	3.5- 4.0 sec	86.1 KBytes	1.41 Mb/s				
[4]	3.5- 4.0 sec	68.9 KBytes	1.13 Mb/s	2.188 ms	0/	48 (0%)	
[5]	4.0- 4.5 sec	86.1 KBytes	1.41 Mb/s				
[4]	4.0- 4.5 sec	97.6 KBytes	1.60 Mb/s	2.399 ms	0/	68 (0%)	
[5]	4.5- 5.0 sec	84.7 KBytes	1.39 Mb/s				
[4]	4.5- 5.0 sec	94.7 KBytes	1.55 Mb/s	1.978 ms	0/	66 (0%)	
[5]	5.0- 5.5 sec	86.1 KBytes	1.41 Mb/s				
[4]	5.0- 5.5 sec	70.3 KBytes	1.15 Mb/s	2.054 ms	0/	49 (0%)	
[5]	5.5- 6.0 sec	86.1 KBytes	1.41 Mb/s				
[4]	5.5- 6.0 sec	108 KBytes	1.76 Mb/s	2.471 ms	0/	75 (0%)	
[5]	6.0- 6.5 sec	84.7 KBytes	1.39 Mb/s				
[4]	6.0- 6.5 sec	60.3 KBytes	988 Kbits/sec	3.895 ms	0/	42 (0%)	
[5]	6.5- 7.0 sec	86.1 KBytes	1.41 Mb/s				
[4]	6.5- 7.0 sec	112 KBytes	1.83 Mb/s	2.314 ms	0/	78 (0%)	
[5]	7.0- 7.5 sec	86.1 KBytes	1.41 Mb/s				
[4]	7.0- 7.5 sec	56.0 KBytes	917 Kbits/sec	3.821 ms	0/	39 (0%)	
[5]	7.5- 8.0 sec	86.1 KBytes	1.41 Mb/s				
[4]	7.5- 8.0 sec	112 KBytes	1.83 Mb/s	4.677 ms	0/	78 (0%)	
[5]	8.0- 8.5 sec	84.7 KBytes	1.39 Mb/s				
[4]	8.0- 8.5 sec	56.0 KBytes	917 Kbits/sec	2.818 ms	0/	39 (0%)	
[5]	8.5- 9.0 sec	86.1 KBytes	1.41 Mb/s				
[4]	8.5- 9.0 sec	113 KBytes	1.86 Mb/s	4.904 ms	0/	79 (0%)	
[5]	9.0- 9.5 sec	86.1 KBytes	1.41 Mb/s				
[4]	9.0- 9.5 sec	56.0 KBytes	917 Kbits/sec	2.433 ms	0/	39 (0%)	
[5]	9.5-10.0 sec	84.7 KBytes	1.39 Mb/s				

Relatório do *Iperf* para quinze chamadas.

[5]	0.0- 0.5 sec	115 KBytes	1.88 Mb/s				
[3]	0.0- 0.5 sec	125 KBytes	2.05 Mb/s	0,582 ms	0/	87 (0%)	
[5]	0.5- 1.0 sec	115 KBytes	1.88 Mb/s				
[3]	0.5- 1.0 sec	113 KBytes	1.86 Mb/s	1,073 ms	0/	79 (0%)	
[5]	1.0- 1.5 sec	113 KBytes	1.86 Mb/s				
[3]	1.0- 1.5 sec	115 KBytes	1.88 Mb/s	0,744 ms	0/	80 (0%)	
[5]	1.5- 2.0 sec	115 KBytes	1.88 Mb/s				
[3]	1.5- 2.0 sec	115 KBytes	1.88 Mb/s	1,134 ms	0/	80 (0%)	
[5]	2.0- 2.5 sec	113 KBytes	1.86 Mb/s				
[3]	2.0- 2.5 sec	113 KBytes	1.86 Mb/s	0,960 ms	0/	79 (0%)	
[5]	2.5- 3.0 sec	115 KBytes	1.88 Mb/s				
[3]	2.5- 3.0 sec	115 KBytes	1.88 Mb/s	0,512 ms	0/	80 (0%)	
[5]	3.0- 3.5 sec	115 KBytes	1.88 Mb/s				
[3]	3.0- 3.5 sec	113 KBytes	1.86 Mb/s	1,013 ms	0/	79 (0%)	
[5]	3.5- 4.0 sec	113 KBytes	1.86 Mb/s				
[3]	3.5- 4.0 sec	115 KBytes	1.88 Mb/s	0,585 ms	0/	80 (0%)	
[3]	4.0- 4.5 sec	115 KBytes	1.88 Mb/s	0,818 ms	0/	80 (0%)	
[5]	4.0- 4.5 sec	105 KBytes	1.72 Mb/s				
[3]	4.5- 5.0 sec	113 KBytes	1.86 Mb/s	1,922 ms	0/	79 (0%)	
[5]	4.5- 5.0 sec	73.2 KBytes	1.20 Mb/s				
[5]	5.0- 5.5 sec	165 KBytes	2.70 Mb/s				
[3]	5.0- 5.5 sec	115 KBytes	1.88 Mb/s	1,131 ms	0/	80 (0%)	
[3]	5.5- 6.0 sec	113 KBytes	1.86 Mb/s	1,563 ms	0/	79 (0%)	
[5]	5.5- 6.0 sec	105 KBytes	1.72 Mb/s				
[5]	6.0- 6.5 sec	123 KBytes	2.02 Mb/s				
[3]	6.0- 6.5 sec	115 KBytes	1.88 Mb/s	1,273 ms	0/	80 (0%)	
[3]	6.5- 7.0 sec	115 KBytes	1.88 Mb/s	1,236 ms	0/	80 (0%)	
[3]	7.0- 7.5 sec	113 KBytes	1.86 Mb/s	0,764 ms	0/	79 (0%)	
[5]	6.5- 7.0 sec	102 KBytes	1.67 Mb/s				
[5]	7.0- 7.5 sec	0.00 Bytes	0.00 b/s				
[5]	7.5- 8.0 sec	241 KBytes	3.95 Mb/s				
[3]	7.5- 8.0 sec	113 KBytes	1.86 Mb/s	2,588 ms	0/	79 (0%)	
[5]	8.0- 8.5 sec	115 KBytes	1.88 Mb/s				
[3]	8.0- 8.5 sec	115 KBytes	1.88 Mb/s	1,786 ms	0/	80 (0%)	
[5]	8.5- 9.0 sec	113 KBytes	1.86 Mb/s				
[3]	8.5- 9.0 sec	115 KBytes	1.88 Mb/s	0,867 ms	0/	80 (0%)	
[3]	9.0- 9.5 sec	115 KBytes	1.88 Mb/s	1,867 ms	0/	80 (0%)	
[5]	9.0- 9.5 sec	81.8 KBytes	1.34 Mb/s				
[3]	0.0-10.0 sec	2.23 MBytes	1.88 Mb/s	2,751 ms	0/	1593 (0%)	

Relatório do *Iperf* para vinte chamadas.

[5]	0.0- 0.5 sec	225 KBytes	3.69 Mb/s				
[4]	0.0- 0.5 sec	144 KBytes	2.35 Mb/s	8,225 ms	0/	100 (0%)	
[5]	0.5- 1.0 sec	228 KBytes	3.74 Mb/s				
[4]	0.5- 1.0 sec	103 KBytes	1.69 Mb/s	4,500 ms	0/	72 (0%)	
[5]	1.0- 1.5 sec	228 KBytes	3.74 Mb/s				
[4]	1.0- 1.5 sec	48.8 KBytes	800 Kbits/sec	6,211 ms	2/	36 (5.6%)	
[5]	1.5- 2.0 sec	228 KBytes	3.74 Mb/s				
[4]	1.5- 2.0 sec	47.4 KBytes	776 Kbits/sec	7,626 ms	0/	33 (0%)	
[5]	2.0- 2.5 sec	227 KBytes	3.72 Mb/s				
[4]	2.0- 2.5 sec	28.7 KBytes	470 Kbits/sec	12,940 ms	1/	21 (4.8%)	
[5]	2.5- 3.0 sec	228 KBytes	3.74 Mb/s				
[4]	2.5- 3.0 sec	44.5 KBytes	729 Kbits/sec	17,357 ms	0/	31 (0%)	
[5]	3.0- 3.5 sec	228 KBytes	3.74 Mb/s				
[4]	3.0- 3.5 sec	109 KBytes	1.79 Mb/s	8,873 ms	0/	76 (0%)	
[5]	3.5- 4.0 sec	228 KBytes	3.74 Mb/s				
[4]	3.5- 4.0 sec	116 KBytes	1.91 Mb/s	9,567 ms	0/	81 (0%)	
[5]	4.0- 4.5 sec	228 KBytes	3.74 Mb/s				
[4]	4.0- 4.5 sec	68.9 KBytes	1.13 Mb/s	10,297 ms	0/	48 (0%)	
[5]	4.5- 5.0 sec	228 KBytes	3.74 Mb/s				
[4]	4.5- 5.0 sec	56.0 KBytes	917 Kbits/sec	12,782 ms	1/	40 (2.5%)	
[5]	5.0- 5.5 sec	228 KBytes	3.74 Mb/s				
[4]	5.0- 5.5 sec	58.9 KBytes	964 Kbits/sec	10,042 ms	1/	42 (2.4%)	
[5]	5.5- 6.0 sec	228 KBytes	3.74 Mb/s				
[4]	5.5- 6.0 sec	48.8 KBytes	800 Kbits/sec	8,780 ms	1/	35 (2.9%)	
[5]	6.0- 6.5 sec	228 KBytes	3.74 Mb/s				
[4]	6.0- 6.5 sec	122 KBytes	2.00 Mb/s	2,751 ms	0/	85 (0%)	
[5]	6.5- 7.0 sec	227 KBytes	3.72 Mb/s				
[4]	6.5- 7.0 sec	123 KBytes	2.02 Mb/s	3,482 ms	0/	86 (0%)	
[5]	7.0- 7.5 sec	228 KBytes	3.74 Mb/s				
[4]	7.0- 7.5 sec	152 KBytes	2.49 Mb/s	4,269 ms	0/	106 (0%)	
[5]	7.5- 8.0 sec	228 KBytes	3.74 Mb/s				
[4]	7.5- 8.0 sec	113 KBytes	1.86 Mb/s	5,015 ms	0/	79 (0%)	
[5]	8.0- 8.5 sec	228 KBytes	3.74 Mb/s				
[4]	8.0- 8.5 sec	172 KBytes	2.82 Mb/s	2,944 ms	0/	120 (0%)	
[5]	8.5- 9.0 sec	228 KBytes	3.74 Mb/s				
[4]	8.5- 9.0 sec	86.1 KBytes	1.41 Mb/s	4,548 ms	0/	60 (0%)	
[5]	9.0- 9.5 sec	228 KBytes	3.74 Mb/s				
[4]	9.0- 9.5 sec	54.6 KBytes	894 Kbits/sec	6,074 ms	1/	39 (2.6%)	
[5]	9.5-10.0 sec	228 KBytes	3.74 Mb/s				
[5]	0.0-10.0 sec	4.45 MBytes	3.73 Mb/s				

Relatório do *Iperf* para quarenta chamadas.

[5]	0.0- 0.5 sec	283 KBytes	4.63 Mbits/sec				
[3]	0.0- 0.5 sec	8.61 KBytes	141 Kbits/sec	22.740 ms	0/	6 (0%)	
[5]	0.5- 1.0 sec	284 KBytes	4.66 Mbits/sec				
[3]	0.5- 1.0 sec	12.9 KBytes	212 Kbits/sec	34.062 ms	0/	9 (0%)	
[5]	1.0- 1.5 sec	286 KBytes	4.68 Mbits/sec				
[3]	1.0- 1.5 sec	11.5 KBytes	188 Kbits/sec	42.987 ms	1/	9 (11%)	
[5]	1.5- 2.0 sec	284 KBytes	4.66 Mbits/sec				
[3]	1.5- 2.0 sec	7.18 KBytes	118 Kbits/sec	58.719 ms	1/	6 (17%)	
[5]	2.0- 2.5 sec	286 KBytes	4.68 Mbits/sec				
[3]	2.0- 2.5 sec	4.31 KBytes	70.6 Kbits/sec	73.377 ms	0/	3 (0%)	
[5]	2.5- 3.0 sec	286 KBytes	4.68 Mbits/sec				
[3]	2.5- 3.0 sec	11.5 KBytes	188 Kbits/sec	66.437 ms	0/	8 (0%)	
[5]	3.0- 3.5 sec	284 KBytes	4.66 Mbits/sec				
[3]	3.0- 3.5 sec	11.5 KBytes	188 Kbits/sec	64.338 ms	1/	9 (11%)	
[5]	3.5- 4.0 sec	286 KBytes	4.68 Mbits/sec				
[3]	3.5- 4.0 sec	1.44 KBytes	23.5 Kbits/sec	67.455 ms	0/	1 (0%)	
[5]	4.0- 4.5 sec	284 KBytes	4.66 Mbits/sec				
[3]	4.0- 4.5 sec	7.18 KBytes	118 Kbits/sec	89.529 ms	1/	6 (17%)	
[5]	4.5- 5.0 sec	286 KBytes	4.68 Mbits/sec				
[3]	4.5- 5.0 sec	5.74 KBytes	94.1 Kbits/sec	95.803 ms	0/	4 (0%)	
[5]	5.0- 5.5 sec	284 KBytes	4.66 Mbits/sec				
[3]	5.0- 5.5 sec	8.61 KBytes	141 Kbits/sec	95.534 ms	0/	6 (0%)	
[5]	5.5- 6.0 sec	286 KBytes	4.68 Mbits/sec				
[3]	5.5- 6.0 sec	10.0 KBytes	165 Kbits/sec	85.313 ms	0/	7 (0%)	
[5]	6.0- 6.5 sec	286 KBytes	4.68 Mbits/sec				
[3]	6.0- 6.5 sec	10.0 KBytes	165 Kbits/sec	79.411 ms	1/	8 (12%)	
[5]	6.5- 7.0 sec	284 KBytes	4.66 Mbits/sec				
[3]	6.5- 7.0 sec	18.7 KBytes	306 Kbits/sec	52.619 ms	0/	13 (0%)	
[5]	7.0- 7.5 sec	286 KBytes	4.68 Mbits/sec				
[3]	7.0- 7.5 sec	23.0 KBytes	376 Kbits/sec	36.959 ms	1/	17 (5.9%)	
[5]	7.5- 8.0 sec	284 KBytes	4.66 Mbits/sec				
[3]	7.5- 8.0 sec	18.7 KBytes	306 Kbits/sec	34.026 ms	0/	13 (0%)	
[5]	8.0- 8.5 sec	286 KBytes	4.68 Mbits/sec				
[3]	8.0- 8.5 sec	7.18 KBytes	118 Kbits/sec	36.944 ms	0/	5 (0%)	
[5]	8.5- 9.0 sec	284 KBytes	4.66 Mbits/sec				
[3]	8.5- 9.0 sec	11.5 KBytes	188 Kbits/sec	56.550 ms	1/	9 (11%)	
[5]	9.0- 9.5 sec	286 KBytes	4.68 Mbits/sec				
[3]	9.0- 9.5 sec	12.9 KBytes	212 Kbits/sec	53.790 ms	0/	9 (0%)	
[5]	9.5-10.0 sec	286 KBytes	4.68 Mbits/sec				

Relatório do *Iperf* para cinquenta chamadas.

[4]	0.0- 0.5 sec	115 KBytes	1.88 Mb/s	5.675 ms	0/	80 (0%)
[5]	0.5- 1.0 sec	342 KBytes	5.60 Mb/s			
[4]	0.5- 1.0 sec	67.5 KBytes	1.11 Mb/s	6.188 ms	1/	48 (2.1%)
[5]	1.0- 1.5 sec	342 KBytes	5.60 Mb/s			
[4]	1.0- 1.5 sec	96.2 KBytes	1.58 Mb/s	7.595 ms	0/	67 (0%)
[5]	1.5- 2.0 sec	343 KBytes	5.62 Mb/s			
[4]	1.5- 2.0 sec	50.2 KBytes	823 Kbits/sec	17.409 ms	0/	35 (0%)
[5]	2.0- 2.5 sec	342 KBytes	5.60 Mb/s			
[4]	2.0- 2.5 sec	109 KBytes	1.79 Mb/s	3.066 ms	0/	76 (0%)
[5]	2.5- 3.0 sec	342 KBytes	5.60 Mb/s			
[4]	2.5- 3.0 sec	112 KBytes	1.83 Mb/s	4.488 ms	0/	78 (0%)
[5]	3.0- 3.5 sec	294 KBytes	4.82 Mb/s			
[4]	3.0- 3.5 sec	129 KBytes	2.12 Mb/s	8.556 ms	0/	90 (0%)
[5]	3.5- 4.0 sec	390 KBytes	6.40 Mb/s			
[4]	3.5- 4.0 sec	96.2 KBytes	1.58 Mb/s	7.399 ms	0/	67 (0%)
[5]	4.0- 4.5 sec	342 KBytes	5.60 Mb/s			
[4]	4.0- 4.5 sec	144 KBytes	2.35 Mb/s	4.285 ms	0/	100 (0%)
[5]	4.5- 5.0 sec	342 KBytes	5.60 Mb/s			
[4]	4.5- 5.0 sec	61.7 KBytes	1.01 Mb/s	15.705 ms	0/	43 (0%)
[5]	5.0- 5.5 sec	343 KBytes	5.62 Mb/s			
[4]	5.0- 5.5 sec	132 KBytes	2.16 Mb/s	4.612 ms	0/	92 (0%)
[5]	5.5- 6.0 sec	342 KBytes	5.60 Mb/s			
[4]	5.5- 6.0 sec	73.2 KBytes	1.20 Mb/s	5.527 ms	0/	51 (0%)
[5]	6.0- 6.5 sec	343 KBytes	5.62 Mb/s			
[4]	6.0- 6.5 sec	58.9 KBytes	964 Kbits/sec	7.680 ms	0/	41 (0%)
[5]	6.5- 7.0 sec	342 KBytes	5.60 Mb/s			
[4]	6.5- 7.0 sec	123 KBytes	2.02 Mb/s	7.086 ms	0/	86 (0%)
[5]	7.0- 7.5 sec	342 KBytes	5.60 Mb/s			
[4]	7.0- 7.5 sec	106 KBytes	1.74 Mb/s	2.859 ms	0/	74 (0%)
[5]	7.5- 8.0 sec	343 KBytes	5.62 Mb/s			
[4]	7.5- 8.0 sec	90.4 KBytes	1.48 Mb/s	10.005 ms	0/	63 (0%)
[5]	8.0- 8.5 sec	342 KBytes	5.60 Mb/s			
[4]	8.0- 8.5 sec	128 KBytes	2.09 Mb/s	8.103 ms	48/	137 (35%)
[5]	8.5- 9.0 sec	342 KBytes	5.60 Mb/s			
[4]	8.5- 9.0 sec	103 KBytes	1.69 Mb/s	10.827 ms	44/	116 (38%)
[5]	9.0- 9.5 sec	342 KBytes	5.60 Mb/s			
[4]	9.0- 9.5 sec	152 KBytes	2.49 Mb/s	10.873 ms	216/	322 (67%)
[5]	9.5-10.0 sec	343 KBytes	5.62 Mb/s			
[5]	0.0-10.0 sec	6.68 MBytes	5.60 Mb/s			

Relatório do *Iperf* para sessenta chamadas.

[5]	0.0- 0.5 sec	457 KBytes	7.48 Mb/s				
[3]	0.0- 0.5 sec	207 KBytes	3.39 Mb/s	6.010 ms	0/ 144 (0%)		
[5]	0.5- 1.0 sec	457 KBytes	7.48 Mb/s				
[3]	0.5- 1.0 sec	268 KBytes	4.40 Mb/s	3.622 ms	0/ 187 (0%)		
[5]	1.0- 1.5 sec	457 KBytes	7.48 Mb/s				
[3]	1.0- 1.5 sec	254 KBytes	4.16 Mb/s	7.870 ms	0/ 177 (0%)		
[5]	1.5- 2.0 sec	457 KBytes	7.48 Mb/s				
[3]	1.5- 2.0 sec	283 KBytes	4.63 Mb/s	0.976 ms	0/ 197 (0%)		
[5]	2.0- 2.5 sec	457 KBytes	7.48 Mb/s				
[3]	2.0- 2.5 sec	248 KBytes	4.07 Mb/s	4.230 ms	0/ 173 (0%)		
[5]	2.5- 3.0 sec	457 KBytes	7.48 Mb/s				
[3]	2.5- 3.0 sec	293 KBytes	4.80 Mb/s	1.916 ms	0/ 204 (0%)		
[5]	3.0- 3.5 sec	455 KBytes	7.46 Mb/s				
[3]	3.0- 3.5 sec	271 KBytes	4.45 Mb/s	1.890 ms	0/ 189 (0%)		
[5]	3.5- 4.0 sec	457 KBytes	7.48 Mb/s				
[3]	3.5- 4.0 sec	251 KBytes	4.12 Mb/s	3.502 ms	0/ 175 (0%)		
[5]	4.0- 4.5 sec	457 KBytes	7.48 Mb/s				
[3]	4.0- 4.5 sec	238 KBytes	3.90 Mb/s	4.716 ms	0/ 166 (0%)		
[5]	4.5- 5.0 sec	457 KBytes	7.48 Mb/s				
[3]	4.5- 5.0 sec	201 KBytes	3.29 Mb/s	5.426 ms	0/ 140 (0%)		
[5]	5.0- 5.5 sec	457 KBytes	7.48 Mb/s				
[3]	5.0- 5.5 sec	243 KBytes	3.97 Mb/s	6.688 ms	0/ 169 (0%)		
[5]	5.5- 6.0 sec	457 KBytes	7.48 Mb/s				
[3]	5.5- 6.0 sec	184 KBytes	3.01 Mb/s	6.281 ms	114/ 242 (47%)		
[5]	6.0- 6.5 sec	457 KBytes	7.48 Mb/s				
[3]	6.0- 6.5 sec	274 KBytes	4.49 Mb/s	11.612 ms	140/ 331 (42%)		
[5]	6.5- 7.0 sec	457 KBytes	7.48 Mb/s				
[3]	6.5- 7.0 sec	204 KBytes	3.34 Mb/s	6.048 ms	118/ 260 (45%)		
[3]	7.0- 7.5 sec	200 KBytes	3.27 Mb/s	2.878 ms	109/ 248 (44%)		
[5]	7.0- 7.5 sec	396 KBytes	6.49 Mb/s				
[5]	7.5- 8.0 sec	0.00 Bytes	0.00 bits/sec				
[3]	7.5- 8.0 sec	159 KBytes	2.61 Mb/s	5.931 ms	194/ 305 (64%)		
[5]	8.0- 8.5 sec	952 KBytes	15.6 Mb/s				
[3]	8.0- 8.5 sec	194 KBytes	3.18 Mb/s	5.960 ms	0/ 135 (0%)		
[3]	8.5- 9.0 sec	223 KBytes	3.65 Mb/s	2.276 ms	269/ 424 (63%)		
[5]	8.5- 9.0 sec	339 KBytes	5.55 Mb/s				
[5]	9.0- 9.5 sec	570 KBytes	9.34 Mb/s				
[3]	9.0- 9.5 sec	240 KBytes	3.93 Mb/s	5.028 ms	197/ 364 (54%)		
[3]	9.5-10.0 sec	185 KBytes	3.03 Mb/s	17.261 ms	233/ 362 (64%)		
[5]	9.5-10.0 sec	428 KBytes	7.01 Mb/s				

Relatório do *Iperf* para oitenta chamadas.

[5]	0.0- 0.5 sec	564 KBytes	9.24 Mbits/sec				
[4]	0.0- 0.5 sec	296 KBytes	4.85 Mbits/sec	1.421 ms	0/	206 (0%)	
[5]	0.5- 1.0 sec	570 KBytes	9.34 Mbits/sec				
[4]	0.5- 1.0 sec	270 KBytes	4.42 Mbits/sec	0.984 ms	0/	188 (0%)	
[5]	1.0- 1.5 sec	570 KBytes	9.34 Mbits/sec				
[4]	1.0- 1.5 sec	187 KBytes	3.06 Mbits/sec	5.392 ms	0/	130 (0%)	
[5]	1.5- 2.0 sec	570 KBytes	9.34 Mbits/sec				
[4]	1.5- 2.0 sec	257 KBytes	4.21 Mbits/sec	2.650 ms	0/	179 (0%)	
[5]	2.0- 2.5 sec	570 KBytes	9.34 Mbits/sec				
[4]	2.0- 2.5 sec	263 KBytes	4.30 Mbits/sec	3.009 ms	0/	183 (0%)	
[5]	2.5- 3.0 sec	570 KBytes	9.34 Mbits/sec				
[4]	2.5- 3.0 sec	184 KBytes	3.01 Mbits/sec	6.247 ms	0/	128 (0%)	
[5]	3.0- 3.5 sec	571 KBytes	9.36 Mbits/sec				
[4]	3.0- 3.5 sec	257 KBytes	4.21 Mbits/sec	5.050 ms	0/	179 (0%)	
[5]	3.5- 4.0 sec	567 KBytes	9.29 Mbits/sec				
[4]	3.5- 4.0 sec	240 KBytes	3.93 Mbits/sec	1.608 ms	0/	167 (0%)	
[5]	4.0- 4.5 sec	573 KBytes	9.38 Mbits/sec				
[4]	4.0- 4.5 sec	205 KBytes	3.36 Mbits/sec	4.196 ms	0/	143 (0%)	
[5]	4.5- 5.0 sec	570 KBytes	9.34 Mbits/sec				
[4]	4.5- 5.0 sec	263 KBytes	4.30 Mbits/sec	10.089 ms	207/	390 (53%)	
[5]	5.0- 5.5 sec	570 KBytes	9.34 Mbits/sec				
[4]	5.0- 5.5 sec	207 KBytes	3.39 Mbits/sec	4.911 ms	187/	331 (56%)	
[5]	5.5- 6.0 sec	570 KBytes	9.34 Mbits/sec				
[4]	5.5- 6.0 sec	238 KBytes	3.90 Mbits/sec	8.707 ms	236/	402 (59%)	
[5]	6.0- 6.5 sec	570 KBytes	9.34 Mbits/sec				
[4]	6.0- 6.5 sec	256 KBytes	4.19 Mbits/sec	3.457 ms	164/	342 (48%)	
[5]	6.5- 7.0 sec	571 KBytes	9.36 Mbits/sec				
[4]	6.5- 7.0 sec	192 KBytes	3.15 Mbits/sec	7.251 ms	221/	355 (62%)	
[5]	7.0- 7.5 sec	570 KBytes	9.34 Mbits/sec				
[4]	7.0- 7.5 sec	294 KBytes	4.82 Mbits/sec	9.383 ms	174/	379 (46%)	
[5]	7.5- 8.0 sec	570 KBytes	9.34 Mbits/sec				
[4]	7.5- 8.0 sec	238 KBytes	3.90 Mbits/sec	5.040 ms	231/	397 (58%)	
[5]	8.0- 8.5 sec	570 KBytes	9.34 Mbits/sec				
[4]	8.0- 8.5 sec	225 KBytes	3.69 Mbits/sec	2.751 ms	213/	370 (58%)	
[5]	8.5- 9.0 sec	570 KBytes	9.34 Mbits/sec				
[4]	8.5- 9.0 sec	228 KBytes	3.74 Mbits/sec	7.421 ms	381/	540 (71%)	
[5]	9.0- 9.5 sec	570 KBytes	9.34 Mbits/sec				
[4]	9.0- 9.5 sec	247 KBytes	4.05 Mbits/sec	10.299 ms	391/	563 (69%)	
[5]	9.5-10.0 sec	570 KBytes	9.34 Mbits/sec				

Relatório do *Iperf* para cem chamadas.