

Benoni de Oliveira Pires
Juliano de Souza

*Laboratório Integrado de Redes de
Computadores*

São José – SC
Setembro / 2008

Benoni de Oliveira Pires
Juliano de Souza

*Laboratório Integrado de Redes de
Computadores*

Monografia apresentada à Coordenação do Curso Superior de Tecnologia em Sistemas de Telecomunicações do Centro Federal de Educação Tecnológica de Santa Catarina para a obtenção do diploma de Tecnólogo em Sistemas de Telecomunicações.

Orientador:

Prof. Dr. Evandro Cantú

Co-orientador:

Prof. M. Jorge Henrique Busatto Casagrande

CURSO SUPERIOR DE TECNOLOGIA EM SISTEMAS DE TELECOMUNICAÇÕES
CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE SANTA CATARINA

São José – SC

Setembro / 2008

Monografia sob o título “*Laboratório Integrado de Redes de Computadores*”, defendida por Benoni de Oliveira Pires e Juliano de Souza e aprovada em 12 de setembro de 2008, em São José, Santa Catarina, pela banca examinadora assim constituída:

Prof. Dr. Evandro Cantú
Orientador

Prof. M. Jorge Henrique Busatto Casagrande
Co-orientador

Prof. Dr. Carlos Barros Montez
Departamento de Automação e Sistemas - UFSC

Dedicamos este trabalho a todos que estão envolvidos no ensino de redes de computadores.

Agradecimentos

Agradeço a Deus mantenedor de todas as coisas e fonte de todas as dádivas.

Agradeço a minha esposa Márcia que me apoiou neste trabalho e a meus filhos Samuel e Isabel, fontes inesgotáveis de alegria.

Benoni de Oliveira Pires

Agradeço primeiramente a Deus, sem o qual nada existiria.

Agradeço a meu pai Nereu Manoel de Souza e a minha mãe Maria de Souza, sem os quais não seria o que sou hoje e só trouxeram o que é bom e certo para se viver, além do apoio incondicional que sempre procuraram dar.

Aos meus irmãos Nereu, Nizan, Quileu, Lizeth e Leonete pelo carinho e compreensão.

A todos os meu amigos que sempre estão perto nos momentos apropriados, sejam bons ou não tão bons.

Juliano de Souza

Resumo

Apresenta-se neste trabalho um laboratório integrado e flexível desenvolvido com o intuito de facilitar o desenvolvimento de atividades práticas de redes de computadores. Este laboratório contempla, na parte física, um arranjo flexível de equipamentos, possibilitando a montagem de diferentes estruturas de conexão em rede. Na parte lógica, o laboratório faz uso de máquinas virtuais, permitindo a alteração flexível da configuração dos protocolos de rede e a realização de teste de serviços e aplicações.

Palavras-Chave: Ensino-Aprendizagem de Redes de Computadores, Laboratório de Redes de Computadores

Abstract

This work presents an integrated and flexible laboratory, which was developed to facilitate the realization of practical activities in computer networks teaching. The hardware of this laboratory includes a flexible set of equipments, which allows for the assembly of different network connection structures. From the point of view of logics, the laboratory utilizes virtual machines, allowing for a flexible change of the configuration of network protocols and for the testing of services and applications.

Keywords: computer network teaching; computer network laboratory;

Sumário

Lista de Figuras

1	Introdução	p. 19
1.1	Motivação	p. 19
1.2	Proposta	p. 20
1.3	Organização do texto	p. 20
2	Ensino-Aprendizagem de Redes de Computadores	p. 21
2.1	Teoria X Prática no Ensino-Aprendizagem de Redes de Computadores	p. 21
2.2	Ensino de Redes de Computadores no CEFET/SC Unidade São José .	p. 25
3	Laboratórios de redes de computadores	p. 27
4	Laboratório Proposto	p. 33
4.1	Estrutura Física	p. 33
4.2	Estrutura Lógica	p. 44
4.2.1	Teoria e Prática no uso de Máquinas Virtuais	p. 46
4.2.2	Tipos e Modelos de Máquinas Virtuais	p. 46
4.2.3	Uso de Máquinas Virtuais no Ensino de Redes de Computadores	p. 49
4.2.4	Uso de Máquinas Virtuais no Laboratório Integrado de Redes de Computadores	p. 50
4.2.5	Modelo de Máquina Virtual usada no Laboratório de Redes de Computadores I	p. 51

5	Cenários de Redes de Computadores	p. 53
5.1	Cenários de Redes Usando os Equipamentos do Laboratório Proposto	p. 53
5.2	Cenários de Redes com Máquinas Virtuais	p. 56
6	Conclusões	p. 61
7	Anexos	p. 63
7.1	Instalação do VMware Server 1.0.4	p. 63
7.2	Configuração do VMware Server 1.0.4	p. 64
7.3	Comunicação entre Computadores Utilizando Linha Discada	p. 66
7.4	Instalação e Configuração da Rede de Roteadores	p. 67
7.4.1	Testes com PING	p. 70
7.4.2	Testes com Traceroute	p. 70
7.5	Teste de protocolos de roteamento em malha de dois roteadores	p. 71
7.5.1	Testes com PING	p. 73
7.5.2	Traçando Rotas com o traceroute	p. 78
7.5.3	Capturando Pacotes com tcpdump	p. 79
7.6	Instalação e Configuração da Rede de Roteadores Usando Máquinas Vir- tuais	p. 81
7.6.1	Testes com PING	p. 92
7.6.2	Traçando Rotas com o traceroute	p. 97
7.6.3	Capturando Pacotes com tcpdump	p. 100
7.7	Configuração de uma Rede Virtual com Acesso a Internet no Laboratório de Redes I	p. 101
	Testes com PING	p. 107
7.7.1	Traçando Rotas com o traceroute	p. 110
7.7.2	Capturando Pacotes com tcpdump	p. 111

7.8	Configuração de uma Malha de Rede Virtual com Acesso a Internet no Laboratório de Redes I	p. 113
7.8.1	Testes com PING	p. 123
7.8.2	Traçando Rotas com o traceroute	p. 133
7.8.3	Capturando Pacotes com tcpdump	p. 134
	Referências	p. 137

Lista de Figuras

1	Configuração do Laboratório de Redes de Computadores.	p. 28
2	Open Lab	p. 30
3	Estrutura Antiga do Laboratório de Redes I.	p. 33
4	Rack Central.	p. 36
5	Rack Direito.	p. 37
6	Rack Esquerdo.	p. 38
7	Visão Geral do Laboratório.	p. 39
8	Disposição das Bancadas e Equipamentos.	p. 39
9	Esquema Elétrico.	p. 40
10	Esquema do Cabeamento de Rede.	p. 40
11	Aula de Redes de Computadores.	p. 41
12	Aula de Redes de Computadores.	p. 41
13	Aula de Redes de Computadores.	p. 42
14	Pontos de Rede.	p. 42
15	Cabeamento.	p. 43
16	Monitor Tipo I	p. 47
17	Monitor Tipo II	p. 47
18	Aplicação com Linha Discada.	p. 54
19	Estrutura Redes de Roteadores.	p. 55
20	Estrutura Interligando Duas Redes de Roteadores.	p. 56
21	Diagrama de Redes de Roteadores.	p. 57

22	Diagrama de Rede Virtual com Acesso a Internet no Laboratório de Redes I.	p. 58
23	Diagrama da Malha de Rede Virtual com Acesso a Internet no Laboratório de Redes I.	p. 59
24	Aplicação com Linha Discada.	p. 66
25	Estrutura Redes de Roteadores.	p. 69
26	Estrutura Interligando Duas Redes de Roteadores.	p. 72
27	Diagrama de Redes de Roteadores.	p. 82
28	Diagrama de Rede Virtual com Acesso a Internet no Laboratório de Redes I.	p. 102
29	Diagrama da Malha de Rede Virtual com Acesso a Internet no Laboratório de Redes I.	p. 114

1 *Introdução*

1.1 *Motivação*

No ensino de redes de computadores a prática de laboratório é um aspecto fundamental. Porém, a falta de laboratórios bem estruturados que atendam de forma satisfatória as necessidades do ensino de redes de computadores é comum. Faltam não só equipamentos, mas também, locais específicos para ministrar as aulas práticas. Há situações em que o ambiente físico é compartilhado com outras disciplinas, impedindo mudanças na estrutura de rede do laboratório, e por consequência, a não realização, ou a realização de experimentos de forma insatisfatória na área de redes de computadores.

Normalmente, a montagem de experiências exige mudanças na topologia da rede local e acesso privilegiado para fazer a configuração do sistema operacional em cada máquina. Por exemplo, em um experimento que visa a implementação de sub-redes interligadas por roteadores, os estudantes necessitam:

- Reestruturar a rede do laboratório formando sub-redes;
- Reconfigurar endereços de rede e outros atributos dos computadores;
- Configurar computadores específicos como roteadores ou reconfigurar os roteadores existentes.

Todas essas mudanças tornam-se onerosas e quase impraticáveis, pois absorveriam tempo real de aprendizagem, já que num laboratório convencional todas as mudanças realizadas na estrutura física e/ou lógica deveriam ser desfeitas para o uso de outras disciplinas.

1.2 Proposta

Diante da dificuldade de ter um laboratório exclusivo para o ensino de redes de computadores que contemple vários cenários de uma determinada disciplina, foi desenvolvido no Laboratório de Redes de Computadores I da Unidade São José - CEFET/SC, um arranjo físico/lógico que atenda as expectativas das disciplinas de Redes de Computadores I, II e III, dos cursos de Tecnologia em Sistemas de Telecomunicações e Técnico em Telecomunicações, além de outras disciplinas na área de informática. O objetivo deste arranjo físico/lógico é permitir que as disciplinas citadas acima possam usar o laboratório de acordo com suas necessidades sem mudanças estruturais e/ou lógicas que interfiram umas nas outras.

No arranjo físico foi concebida uma estrutura flexível que permite mudanças na topologia de redes do laboratório bem como a criação de vários cenários de redes. Todas estas mudanças podem ser feitas rapidamente sem ocupar o tempo reservado para o ensino, mas fazendo parte do próprio ensino.

No arranjo lógico contemplamos o uso de Máquinas Virtuais (MV). A tecnologia das MV permite a cada estudante construir sua própria rede virtual, criar sub-redes, testar serviços e outras atividades afins, sem interferir na estrutura física e lógica do laboratório.

No desenvolvimento deste projeto será abordada a relação teoria/prática no processo ensino-aprendizagem em redes de computadores, a análise de alguns modelos de laboratórios de redes existentes, bem como a descrição do laboratório proposto, para o qual será detalhado o arranjo de equipamentos que permitam realizar experimentos e o uso de máquinas virtuais.

1.3 Organização do texto

Este trabalho é composto de sete capítulos, incluindo este. O capítulo 2 apresenta a relação entre o ensino teórico e prático em redes de computadores considerando a opinião de autores da área; o capítulo 3 apresenta alguns modelos de laboratórios de redes de computadores; o capítulo 4 descreve o laboratório proposto, sua estrutura física e lógica; o capítulo 5 apresenta cenários criados usando a estrutura física do laboratório e com o uso de máquinas virtuais; o capítulo 6 conclui a monografia, apontando os benefícios alcançados e os trabalhos futuros relacionados à proposta e sua implementação e finalmente o capítulo 7 apresenta os resultados dos testes realizados.

2 Ensino-Aprendizagem de Redes de Computadores

Antes de se falar em um laboratório adequado ao ensino de redes de computadores é preciso fazer uma relação dos principais conceitos necessários para o entendimento desta área, bem como, enumerar alguns tópicos a serem desenvolvidos em laboratório para facilitar a compreensão destes conceitos. Neste capítulo, busca-se, justamente, fazer a relação da parte teórica com o que pode ser feito na parte prática. Diante disto, percebe-se a divisão do ensino de redes de computadores, normalmente realizada em duas abordagens: conceitual e prática. Nesta abordagem, são passados os conceitos em sala de aula, para depois serem vivenciados com experiências em laboratório (KUROSE, 2002).

A seguir, apresenta-se as opiniões de alguns autores sobre a relação teoria-prática no ensino-aprendizagem de Redes de Computadores.

2.1 Teoria X Prática no Ensino-Aprendizagem de Redes de Computadores

Uma boa discussão sobre a relação teoria e prática no ensino-aprendizagem de redes de computadores pode ser encontrada em (KUROSE et al., 2002), que sintetiza a conferência do SIGCOMM (Special Interest Group on Data Communications) realizada de 19 a 23 agosto de 2002, cujo tema foi *Workshop on Computer Networking: Curriculum Designs and Educational Challenges*. Devido a importância do tema é bastante oportuno a exposição das idéias de alguns autores para uma melhor compreensão da importância do ensino prático em redes de computadores.

Abraham (2002), da Universidade do Texas, apresentou seu modelo de curso, o qual utiliza como referência o livro de redes de computadores escrito por Andrew Tanenbaum. Segundo o autor, os estudantes que faziam o referido curso não gostavam da parte pesada de programação, por isso pensaram em diminuir a quantidade de horas dedicadas à

programação e colocar mais aulas práticas no sentido de configuração de redes, utilizando roteadores e switches. Abraham (2002) ainda encontrava algumas dificuldades para ministrar determinados conteúdos, como o caso das camadas do sistema OSI. Para o autor, um dos problemas no ensino de redes é a falta de tempo para cobrir todas as camadas OSI em um semestre. Ele só conseguia cobrir as camadas 1 a 4 e por isso estava pensando em mudar sua abordagem para as camadas do protocolo TCP/IP. Outro problema referido era a falta de um laboratório permanente de redes de computadores, pois cada vez que ia ministrar aulas tinha que reunir todos os componentes necessários de hardware e software e reservar uma sala.

O curso, ora descrito, foi pensado com uma estrutura contendo como introdução tópicos de comunicação de dados, tendo incluso controle de fluxo, comutação de pacotes, topologias de rede, protocolos, Internet e segurança de dados. Eram estudados, também, exemplos de redes já existentes e suas arquiteturas. Além disso, os estudantes recebiam grande experiência prática, já que poderiam estabelecer comunicação entre computadores e entre redes. Estava previsto 60% do tempo do curso dedicado à parte teórica e 40% dedicado à parte prática (ABRAHAM, 2002).

Al-Shaer e Brewster (2002), da Universidade de Chicago, desenvolveram seu currículo para cursos voltados para a área de redes de computadores dividido em duas partes. A primeira seria um estudo geral sobre telecomunicações, focando em comunicação de voz e tecnologia em telefonia; a segunda estaria focada na parte de tecnologia e aplicações da Internet. Nesta última parte seria mais exigido programação em C e C++, mas para isso seria necessária a presença de pré-requisitos como programação em Java, matemática discreta, circuitos eletrônicos, estatística, arquitetura computacional e sistemas operacionais. O curso sugerido abrangeria duas áreas: a primeira envolve sistemas básicos de telecomunicações, incluindo neste contexto regulamentação, meios de transmissão, sinais, técnicas de modulação e codificação e estudo da parte de telefonia; e a segunda parte que envolve a comunicação e transporte de dados, com conceitos de topologias de rede, protocolos de acesso ao meio, roteamento e aplicações de rede. Afirmam os autores que estes cursos devem ser totalmente ministrados com aulas em laboratório, já que é necessária uma grande experiência prática.

Burroughs (2002), da Universidade Estadual de Humboldt, divulgou através de seu artigo como era o curso do qual participava e qual a estrutura oferecida a este pela universidade. Segundo a referida autora, na Universidade Estadual de Humboldt há dois cursos: um de telecomunicações e outro de desenvolvimento e implementação de redes.

No ano de 2000, a universidade adquiriu um laboratório para o ensino de Internet. Depois desta aquisição, ficou mais fácil para fazer um cenário mais próximo do que realmente se pretendia para o ensino de redes. Através de ajuda da instituição, foi possível adicionar elementos à estrutura do laboratório, como ar condicionado, hubs, switches e patch panels. Com isso, o ensino de redes ficou ainda mais fácil, já que proporcionou a possibilidade de realização de um maior número de experiências. De fato, pode-se perceber o quão é importante um laboratório deste gênero, segundo a opinião da autora, quando questiona se os tópicos desta modalidade de ensino podem ser abrangidos sem algum tratamento preliminar ou posterior à teoria. Da mesma forma, questiona se há experiências que podem ser apresentadas e preparadas para suportar a demanda. E, se sim, se há um laboratório com recursos para isso, pois os estudantes necessitam explorar o modo “mãos à obra” de ensino, ou seja, aprender com prática.

Ainda cabe salientar que o referido curso teve sua parte teórica reduzida para 1/3 depois da implementação do laboratório, demonstrando a importância da atividade prática no ensino de redes de computadores.

Em se tratando desta relação entre teoria e prática no ensino de redes de computadores, Comer (2002) demonstrou sua posição ao descrever um currículo para um curso na área. Primeiramente, o autor faz uma separação entre os cursos de graduação e os cursos de pós-graduação. Através do que mostra Comer, um curso de graduação tem como objetivo conhecer de uma maneira mais ampla os conceitos de redes de computadores. O estudante deve ser familiarizado com todos os aspectos que envolvem esta área, mas sem a necessidade de um aprofundamento neste contexto. Quanto aos cursos de pós-graduação, o autor mostra que o objetivo de um curso de pós-graduação é a sua especialização e sua profundidade. O estudante deverá se familiarizar com toda a literatura e ter o entendimento de assuntos mais complexos envolvendo redes de computadores e suas mais recentes tecnologias. Segundo Comer, um curso de redes de computadores para a graduação deverá conter o seguinte currículo:

- uma descrição preliminar sobre redes de computadores e comunicação entre redes de computadores;
- programação voltada para redes de computadores;
- interligação de redes de computadores;
- desenvolvimento de sistema em redes de computadores;

- tecnologias da Web;
- atualidades e tendências em redes de computadores.

Após mencionar estes tópicos, o autor ainda comenta sobre a importância de atividades práticas no ensino de redes de computadores, ao afirmar que “laboratórios formam um aspecto absolutamente essencial para qualquer currículo em redes de computadores porque os estudantes aprendem fazendo. Os laboratórios reforçam os conceitos apresentados em sala de aula, expondo os estudantes a tecnologias práticas, incentivando os mesmos a se interessarem e a apreciarem os detalhes, e manter os cursos próximos à realidade.” Neste mesmo tema, o autor continua seu pensamento afirmando que algumas instituições parecem impedidas pelos custos para poder tirar o proveito de bons laboratórios. Felizmente, muitas escolas estão podendo ter um laboratório com o mínimo necessário, que é um conjunto de computadores em uma LAN conectada à Internet através de um servidor NAT.

Kurose (2002) sugeriu um currículo para cursos básicos de redes de computadores. Tal idéia foi retirada por consenso da conferência já citada anteriormente (KUROSE et al., 2002).Segue a estrutura de tópicos:

- **Rede física**

- Canais digitais;
- Erros e detecção de erros;
- Entender pelo menos um protocolo de acesso ao meio (ex. CSMA)
- Algo sobre redes sem fio (wireless LANs)
- Limites de Shannon e Nyquist

- **Conceitos**

- Comutação de circuitos X Comutação de pacotes;
- Enquadramento e encapsulamento;

- **Interconexão de rede**

- Movendo pacotes através de múltiplas redes (roteamento e inter-rede);
- Endereçamento e encaminhamento;

- **Protocolos**

- O que é um protocolo e como especificá-lo;
- Protocolos de janelas deslizantes para transferência de dados confiável (incluindo TCP);
- Controle de congestionamento;
- **Discussões sobre:**
 - Programação de aplicações cliente/servidor;
 - Programação de soquetes;
 - Gerência e configuração remota de dispositivos;
 - Protocolos de aplicação usados no momento e como eles funcionam;
- **Outros temas:**
 - Segurança como realidade do dia a dia;
 - Criptografia como solução;
 - Elementos de performance (atraso de propagação e transmissão);

O autor complementa sua sugestão fazendo uma crítica, afirmando que nos cursos de redes de computadores, tradicionalmente, não há um ensino prático suficiente aos estudantes para que estes possam manusear os equipamentos de rede e ter acesso a seus softwares. Existe ainda uma certa ausência de atividades práticas em muitos cursos, sendo estes muito presos aos conceitos em sala de aula, o que atrela ao ensino de redes de computadores um certo ar de abstração.

2.2 Ensino de Redes de Computadores no CEFET/SC Unidade São José

No Curso Superior de Tecnologia em Sistemas de Telecomunicações existe uma cadeia de disciplinas que abrangem os tópicos sugeridos por Kurose (2002).

- **Redes de Computadores I:** são abordados os conhecimentos básicos sobre redes de computadores e Internet. São estudados os conceitos de camadas de rede e os seus respectivos protocolos com ênfase na arquitetura TCP/IP, além de introduzir o conceito de redes locais e endereços físicos.

- **Redes de Computadores II:** são estudados os equipamentos de rede em geral, como modems, switches, hubs e roteadores. Aborda-se o seu manuseio e configuração para a constituição da parte de camada física e de enlace do modelo Ethernet.
- **Redes de Computadores III:** é feita uma abordagem sobre protocolos de roteamento e também IPv6, IP móvel e multicast.
- **Redes Multimídia:** são estudadas aplicações multimídias em redes, o tratamento da qualidade de serviço (QoS) e telefonia sobre IP.
- **Gerência de Redes:** é abordado segurança e administração de rede, bem como, servidores de rede em geral.

Fazendo uma análise sobre a comparação entre teoria e prática descrita na sessão 2.1 pode-se concluir que a teoria é fundamental para que se possa colocar as idéias em prática de uma maneira ordenada e ciente. Em redes de computadores é essencial que se tenha uma constante atividade prática para que o acadêmico desperte interesse por estar colocando em prática aquilo que foi discutido em sala de aula. Isso é fundamental para as disciplinas citadas neste tópico, já que estão em sintonia com o que recomendam os autores mencionados anteriormente. Portanto, é de fundamental importância a construção de um ambiente propício para por em prática os conhecimentos trabalhados nestas disciplinas.

3 *Laboratórios de redes de computadores*

Os cursos introdutórios de redes de computadores têm a tendência de ensinar conceitos de um modo bastante abstrato, fazendo com que exista certa dificuldade na compreensão de como estes conceitos são aplicados em uma rede operacional. Por isso, deve existir uma estreita relação entre o ensino teórico e sua aplicação prática. As aulas de laboratórios são essenciais em qualquer currículo de rede de computadores, pois os estudantes aprendem fazendo. As aulas de laboratório reforçam os conceitos apresentados em sala de aula, expõem os estudantes às tecnologias e permitem que apreciem e entendam detalhes.

Abaixo, são relacionados alguns exemplos de laboratórios de redes de computadores apresentados na conferência do SIGCOMM (KUROSE et al., 2002) descrito na sessão 2.1.

Aburdene (2002) descreve um laboratório estruturado para permitir o estudo de tecnologias de hardware e software em redes de computadores e sistemas de computação, com mecanismos de flexibilização que permitem aos alunos fazerem experiências nestas áreas. O laboratório é composto por 14 computadores conectados a rede acadêmica da universidade e três redes privadas experimentais: Fast Ethernet, Gigabit Ethernet e Rede wireless. As redes experimentais possuem vários equipamentos, como switches, roteadores e central de telefonia IP. Cada computador possui uma interface de rede para cada uma das conexões de rede como mostrado na figura 1.

As redes experimentais são conectadas a um roteador *firewall*, que por sua vez está ligado a rede IP do campus. A configuração deste roteador permite a filtragem do tráfego das redes experimentais e proporciona segurança e gerenciamento do tráfego para o resto da rede do campus. A organização da rede de computadores do laboratório pode ser reconfigurada para acomodar novas tecnologias e arquiteturas de rede. Sendo que as tecnologias de redes de computadores evoluem, novas redes experimentais poderão ser adicionadas ao laboratório. Em termos lógicos, cada computador possui um boot duplo, com Windows e Linux Red Hat.

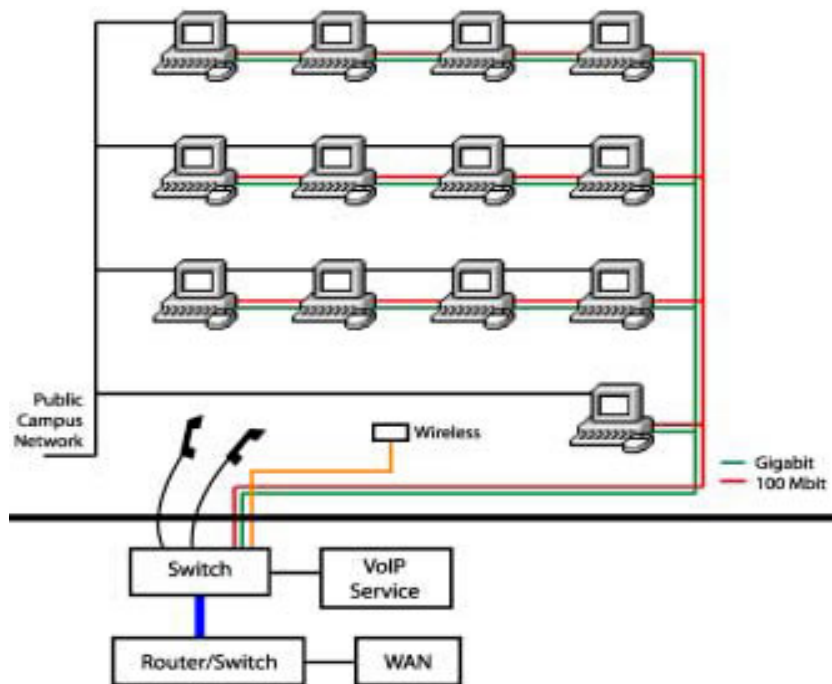


Figura 1: Configuração do Laboratório de Redes de Computadores.

Burroughs (2002) descreve um laboratório estruturado a partir do projeto ITL (Internet Teaching Laboratory) o qual recebeu doações de roteadores da Cisco, possuindo ferramentas e equipamentos de rede que permitem a montagem de diversos cenários de redes de computadores. Cada estação possui *boot* duplo, um conectado a rede acadêmica e outro conectado a rede experimental do laboratório, proporcionando isolamento. Alguns experimentos realizados são:

- Reconhecimento dos equipamentos de rede (interfaces de rede), verificação da configuração da rede, uso do *ping* para testar comunicação entre hosts;
- Identificação dos pinos de um conector RJ45, categoria dos cabos de rede, diferenciação entre categorias, uso de fibra;
- Roteamento e tabela ARP;
- Análise de uma LAN (classe da rede, máscara, endereço da rede);
- Gerenciamento de rede e protocolo SNMP

Cigas (2002) apresenta um laboratório com 16 computadores, cinco servidores, cinco roteadores e cinco PCs com duas interfaces para montar roteadores, além de *hubs* e *switches*. Os experimentos realizados são:

- Experimentos com ping no endereço de *loop back*, na rede local e endereços inexistentes;
- Exame da cache ARP a partir de ações de ping;
- Configuração de quatro computadores para utilizar endereços IP de diferentes redes, ainda que na mesma rede física compartilhada.
- Configuração de servidor DHCP ainda na mesma rede física compartilhada;
- Exame de tráfego TELNET, FTP e HTTP e de tráfego criptografado usando SSH, SFTP e HTTPS;
- Criação de uma LAN isolada;
- Configuração de um roteador para conectar a LAN ao *backbone* usando RIP;
- Configuração de um servidor DNS;
- Construção de um roteador usando Linux;
- Configuração de roteadores usando RIP e OSPF;

Comer (2002) apresenta o laboratório *Xinu*, o qual existe desde 1984, possuindo 20 computadores, vários roteadores, além de hubs e switches. Os exercícios incluem programação de rede usando soquetes, análise de tráfego de rede usando *hub* e *switches*, análise de protocolos na qual os alunos analisam, por exemplo, fragmentação de IP e traçam conexões TCP.

Zarki (2002) utiliza um Laboratório Aberto (*Open Lab*), localizado em uma área pública da universidade, no qual os alunos desenvolvem os experimentos sem supervisão. Os equipamentos do Open Lab estão montados em um rack, como mostrado na figura 2, e consiste em quatro PCs Linux, quatro roteadores e quatro *hubs* Ethernet. Os PCs e os roteadores são controlados por um único teclado e monitor, o qual é conectado a um chaveador de monitor/teclado (KVM). O laboratório não é conectado a Internet, podendo ser facilmente duplicado. Com o Open Lab os alunos completam um total de oito laboratórios em 10 semanas de aula. Os tópicos incluem:

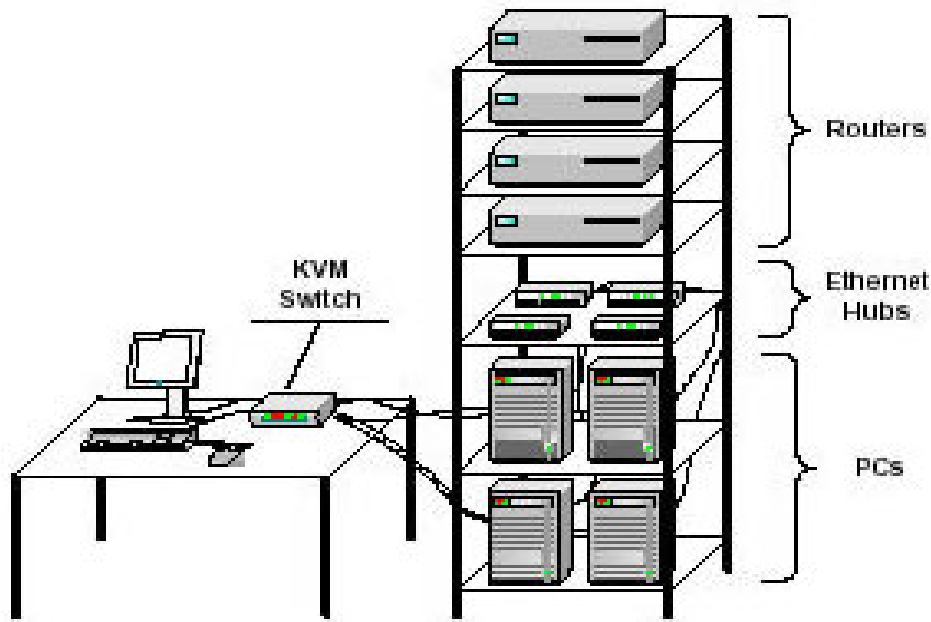


Figura 2: Open Lab

- segmentação de sub-redes;
- roteamento estático;
- protocolos de roteamento;
- LAN com *switch*;
- TCP e UDP, *Multicast*, NAT, DHCP, DNS e SNMP;

Os exercícios consistem de medidas de tráfego e ferramentas de análise de protocolos. Cada laboratório é estruturado em três fases:

- **Pré-laboratório**(primeira fase) - na qual os estudantes leem o material e respondem questões, visando preparar os exercícios práticos.
- **exercícios do laboratório**(segunda fase)- na qual os estudantes trabalham nos equipamentos do Open Lab, seguindo instruções passo a passo do manual do laboratório.
- **relatório do laboratório**(terceira fase)- onde os estudantes analisam os dados obtidos.

O material do laboratório consiste em:

1. Introdução: Revisão dos equipamentos e do Linux;

2. Rede não segmentada: Ethernet, transmitindo e recebendo pacotes, ARP, IP e ICMP;
3. Redes com múltiplos segmentos: Encaminhamento de IP, PC e roteadores comerciais, configuração de roteadores, roteamento estático;
4. Roteamento dinâmico: roteamento estático x dinâmico, RIP, OSPF;
5. LAN chaveadas: Pontes transparentes, algoritmo spanning tree;
6. Protocolos TCP e UDP;
7. Tópicos avançados: *Multicast*, NAT e DHCP; DNS

Além desses exemplos de laboratórios, foi realizada uma visita ao SENAC/SC¹, o qual possui um laboratório de redes de computadores dedicado aos cursos profissionalizantes da Cisco (CCNA e CCNP).

Estes exemplos de laboratório foram o ponto de partida para o desenvolvimento do Laboratório Integrado de Redes de Computadores que será descrito no próximo capítulo.

¹SENAC/SC Avenida Osmar Cunha, Florianópolis - SC.

4 *Laboratório Proposto*

Neste capítulo será apresentado o laboratório proposto neste trabalho.

4.1 Estrutura Física

Com o que foi exposto nos capítulos anteriores, pode-se perceber a grande importância de um laboratório que facilite o ensino de redes de computadores.

Com base nas sugestões colocadas pelos diversos autores da área, conjugado aos exemplos de laboratórios pesquisados e que já se encontram em funcionamento, pensou-se na implementação de um projeto deste gênero para o Centro Federal de Educação Tecnológica de Santa Catarina (CEFET/SC) - Unidade de São José.

Para a elaboração deste projeto optou-se por utilizar o laboratório de Redes de Computadores, o qual era, então, utilizado pelos professores da área de redes de computadores em suas aulas.

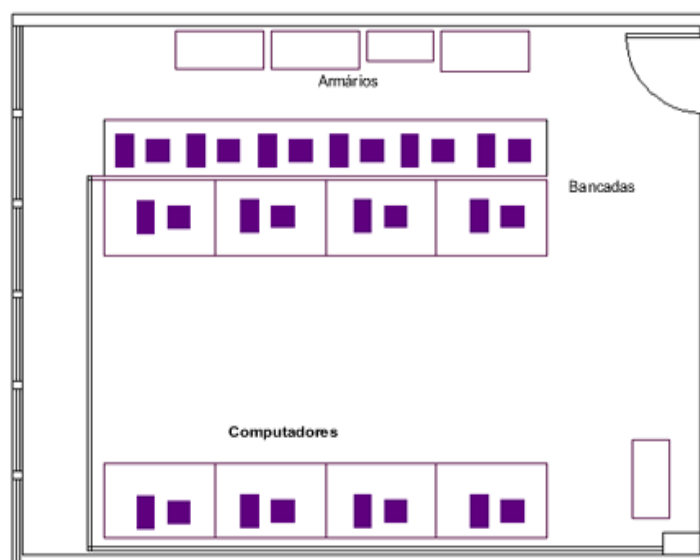


Figura 3: Estrutura Antiga do Laboratório de Redes I.

Nesta época, o referido laboratório oferecia uma estrutura de 14 computadores, sobre bancadas distribuídas de maneira perpendicular à lousa e à mesa do professor, onde o professor ministrava sua aula, como observado na figura 3.

Não haviam equipamentos já previamente instalados em *rack*, todos os pontos de rede do laboratório estavam conectados a um único *patch panel* de onde tinham a conexão com um *hub*. Deste último, possuía-se a comunicação com a rede do CEFET/SC.

Os equipamentos para a realização de experiências, tal como roteadores, modems e outros, ficavam guardados em armários. Quando era necessária a sua utilização, era preciso que os alunos instalassem tanto a parte física quanto a lógica destes equipamentos, o que empreendia grande parte do tempo de aula para isso. A configuração dos computadores era alterada e demandava tempo, posteriormente, para voltar a configuração original. Além disso, o tempo também era utilizado para a desmontagem dos equipamentos e para guardá-los de volta aos armários. Ou seja, grande parte, talvez muitas vezes, metade do tempo de aula era dedicado para montagem, configuração, reconfiguração e desmontagem dos equipamentos e computadores do laboratório.

Pensando nisso e também numa busca de melhoria da qualidade do ensino aprendizagem de redes de computadores, foi desenvolvido o Laboratório Integrado de Redes de Computadores. A idéia central do laboratório de redes de computadores é permitir a montagem de diferentes estruturas de rede, envolvendo equipamentos diversos. Entretanto, foi imposto como requisito que a montagem destas estruturas não poderia prejudicar o funcionamento normal do laboratório. Isto é, após uma aula prática os computadores do laboratório deveriam voltar a operar normalmente, conectados à rede local do CEFET/SC e à Internet.

Para viabilizar a realização destas manobras, construiu-se um arranjo físico de equipamentos, envolvendo duas bancadas, provendo acesso para 24 estações de trabalho, três armários de telecomunicações (*racks*) para abrigar equipamentos de rede, além de cabeamento estruturado específico. Para um aproveitamento eficiente das aulas, pensou-se em mudar a disposição dos móveis do laboratório. Chegou-se então à estrutura que será demonstrada a seguir.

Para cada estação de trabalho disponibilizou-se junto às bancadas dois pontos de rede, um para acesso à rede local do CEFET-SC e à Internet, denominado Rede Externa (RE) e outro para acesso à rede experimental interna do laboratório, denominado Rede Interna (RI). Os 24 pontos de rede da Rede Externa foram montados em um *patch panel* de 24 portas, com conectores padrão RJ45, situado em um Rack Central, e conectados a um *switch* principal que dá acesso à rede local do CEFET/SC e à Internet. Os pontos da

Rede Externa são considerados pontos “sempre operantes”, sendo necessário, portanto, após cada aula prática, simplesmente voltar a plugar cada computador no ponto RE para os mesmos voltarem a operar normalmente. Os pontos da Rede Interna, por sua vez, são utilizados para a montagem de experimentos específicos de redes de computadores. Estes 24 pontos foram montados em dois armários de telecomunicações, denominados Rack Esquerdo e Rack Direito, respectivamente. Em cada um destes racks os pontos de rede estão acessíveis em *patch panels*, permitindo, com manobras rápidas, realizar a montagem de diferentes estruturas.

Para permitir as manobras de conexão entre os diferentes equipamentos, distribuídos e instalados nos três *racks*, 12 pontos de cada *rack* foram espelhados nos demais *racks* e vice-versa. Ainda para facilitar as manobras para conexão dos equipamentos, algumas portas de acesso e interfaces dos equipamentos de telecomunicações foram também montadas nos *patch panels*. Por exemplo, para os roteadores Cisco 1750, estão acessíveis no *patch panel* a interface Ethernet (Eth0), a porta da console (Con) e a interface auxiliar (Aux). Para os modems estão acessíveis as interfaces RX e TX de cada um deles. As figuras 4, 5 e 6 mostram mais detalhes de equipamentos cujas interfaces estão acessíveis nos *patch panels* dos três *racks*.

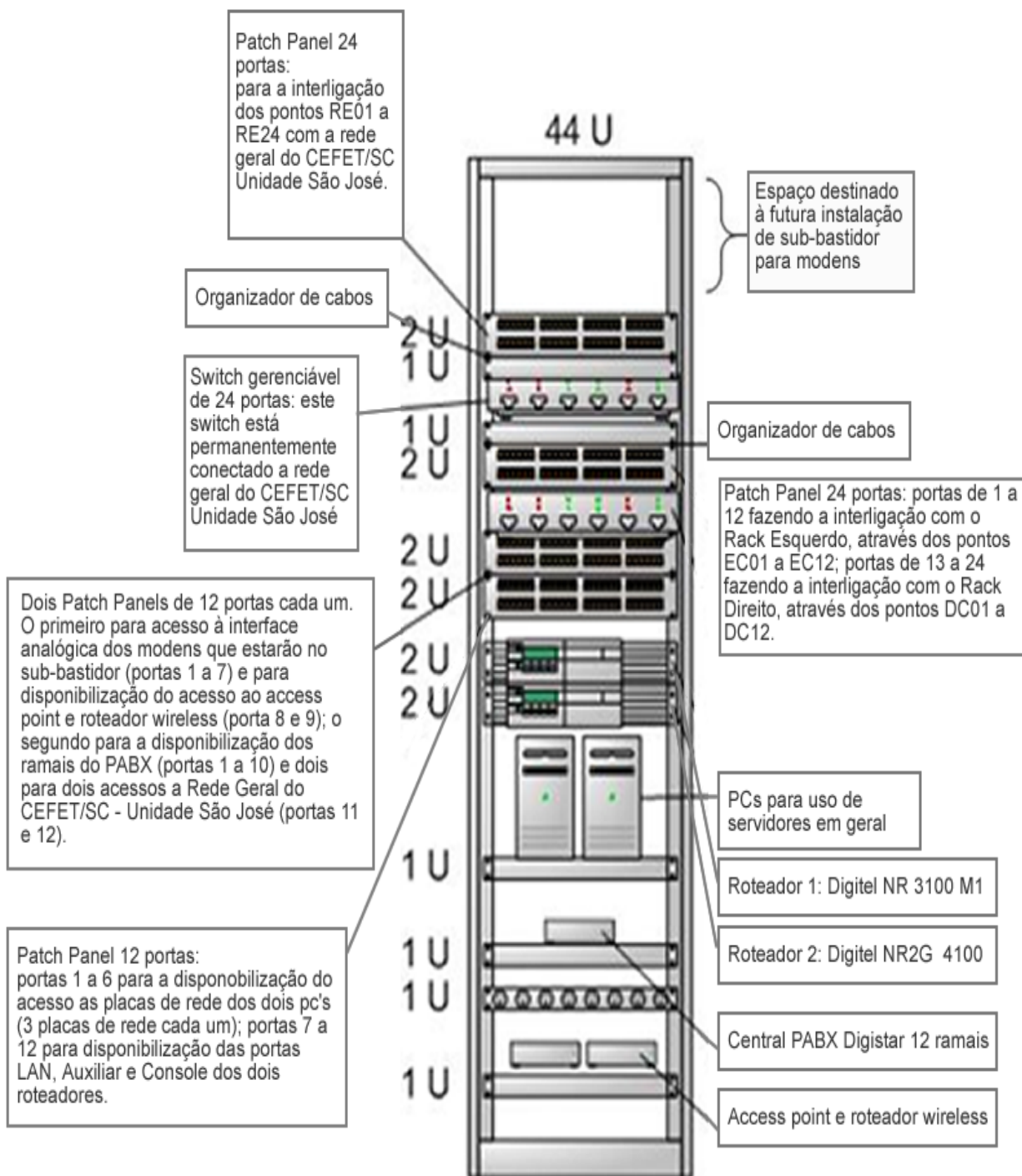


Figura 4: Rack Central.

Uma codificação intuitiva utilizando cores facilita na identificação e acesso de cada ponto, além da utilização de cabos de manobras, também estes diferenciados por cores.

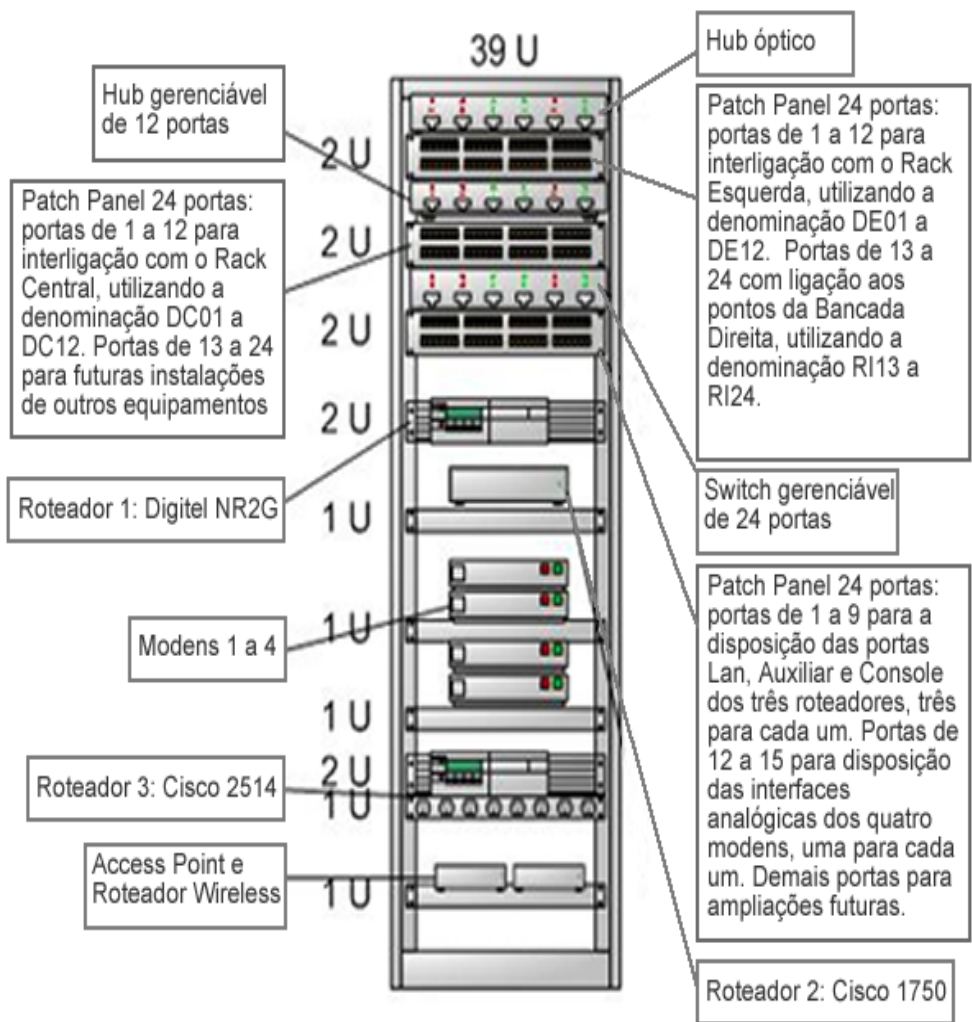


Figura 5: Rack Direito.

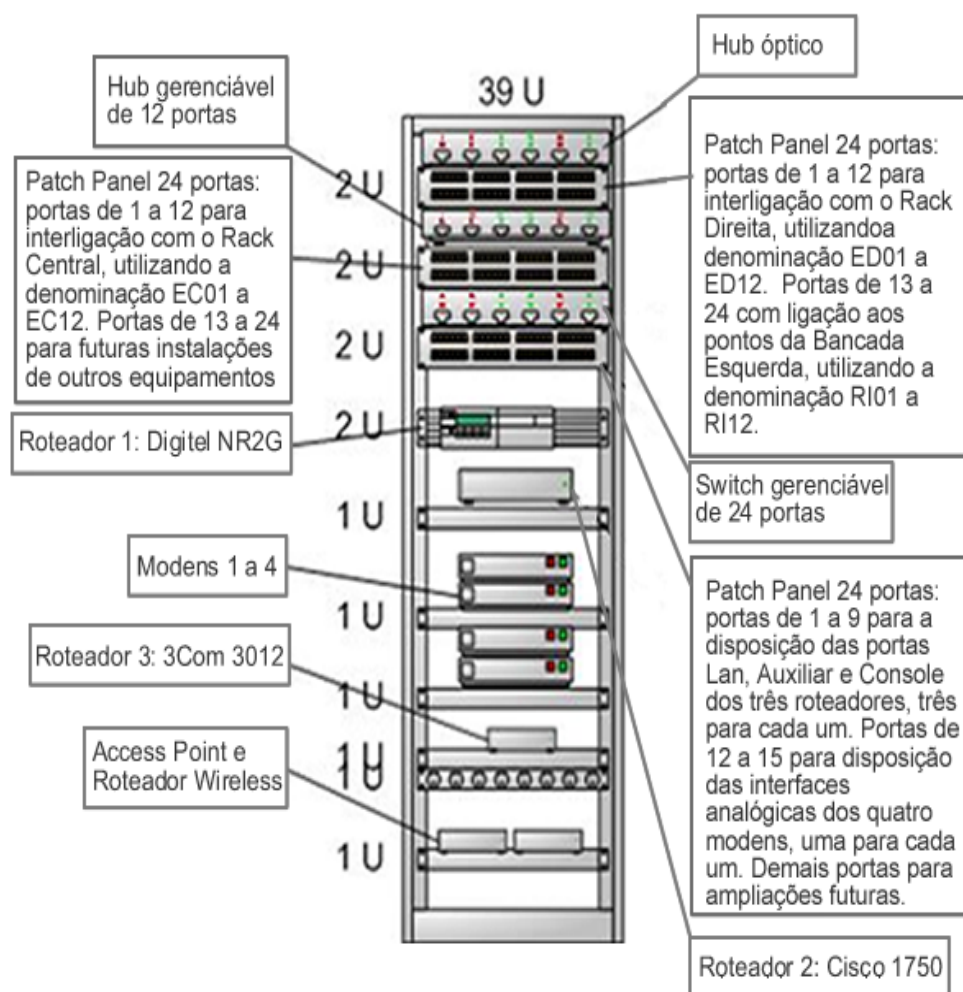


Figura 6: Rack Esquerdo.

Na figura 7 observa-se estrutura do Laboratório Proposto, na figura 8 a disposição das bancadas e equipamentos. Na figura 9 observa-se um esquema elétrico das bancadas e na figura 10 um esquema das conexões de rede do laboratório.

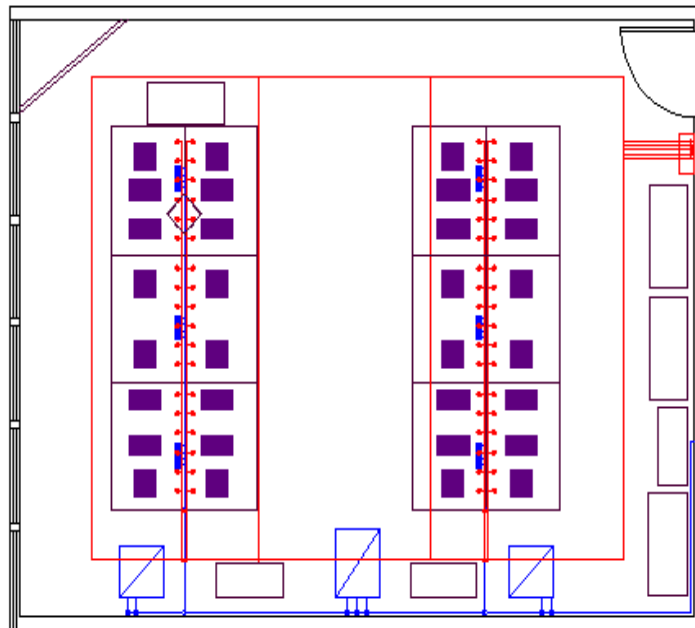


Figura 7: Visão Geral do Laboratório.

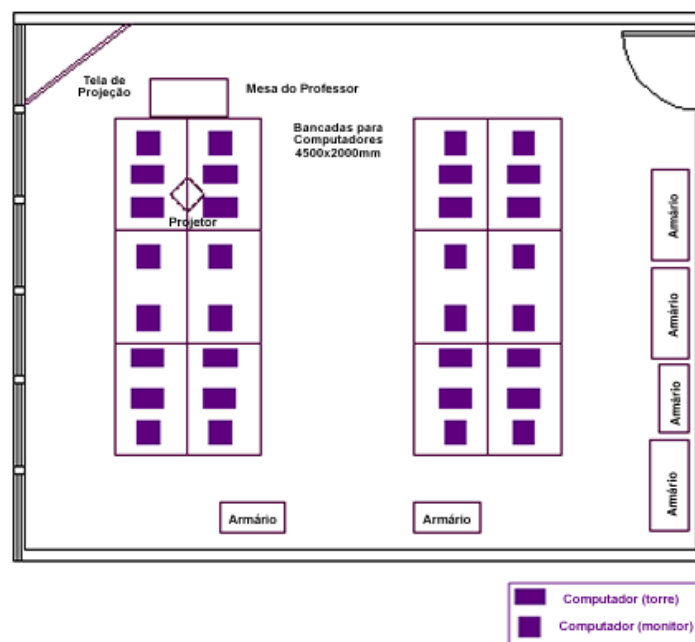


Figura 8: Disposição das Bancadas e Equipamentos.

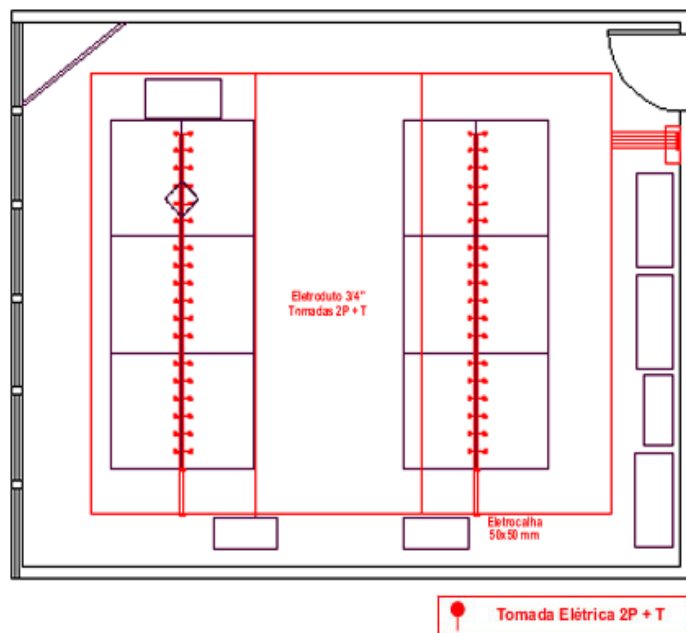


Figura 9: Esquema Elétrico.

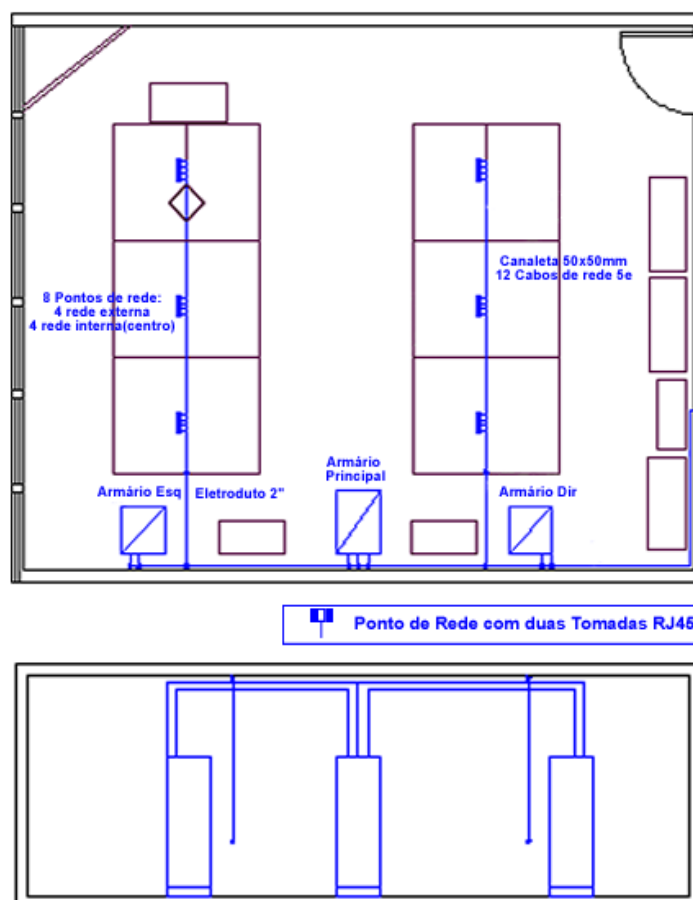


Figura 10: Esquema do Cabeamento de Rede.

A seguir, da figura 11 à 15, algumas fotos de como se encontra atualmente o laboratório.



Figura 11: Aula de Redes de Computadores.



Figura 12: Aula de Redes de Computadores.



Figura 13: Aula de Redes de Computadores.

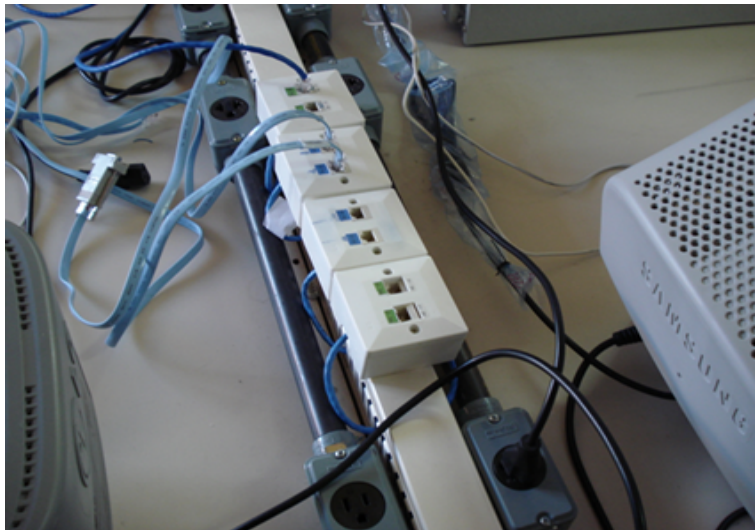


Figura 14: Pontos de Rede.



Figura 15: Cabeamento.

4.2 Estrutura Lógica

A estrutura lógica do Laboratório de Redes de Computadores I atualmente compreende o uso de “máquinas virtuais”. O laboratório é utilizado para aulas das disciplinas do Curso Superior de Tecnologia em Sistemas de Telecomunicações e do Curso Técnico de Telecomunicações com Ênfase em Redes de Computadores, a saber:

- Redes de Computadores I e Sistemas Operacionais - utilizam o Sistema Operacional (SO) Mandriva Linux 2008 sem a necessidade de mudanças na configuração do mesmo;
- Redes de Computadores II - utiliza o SO Windows XP e Mandriva Linux 2008 rodando em máquina virtual VMware;
- Redes de Computadores III - utiliza o SO Mandriva Linux 2008 rodando em máquina virtual VMware ou máquina virtual UML;
- Microcontroladores - utiliza o SO Windows XP rodando em máquina virtual VMware;

O laboratório possui dezesseis (16) computadores ¹ alocados em quatro bancadas e um (01) computador de uso do professor, três (03) armários de telecomunicações (*racks*) para abrigar equipamentos de rede, além de cabeamento estruturado específico.

O sistema operacional (SO) utilizado é o Mandriva Linux 2008 (chamado “anfitrião” devido o uso de máquinas virtuais). A escolha deu-se por ser um software livre² e atender as necessidades das disciplinas ministradas. Nenhuma alteração (tabela de rotas, IPs, arquivos de configuração de rede, entre outras), deve ser realizada no SO anfitrião; desta forma, quaisquer disciplinas poderão ser ministradas sem problemas. A senha de super-usuário (*root*) não é disponibilizada evitando-se desta forma, quaisquer alterações na configuração do sistema operacional e instalação/remoção de software. A fim de dar permissão de acesso as ferramentas *tracert*, *tcpdump*, *ssh*, *arp*, *ip* e *ifconfig* ao usuário “aluno”, foi criado um *script* que estabelece privilégios de super-usuário a este.

¹Cada computador tem como configuração básica de hardware, um disco rígido de 80 GB, memória RAM de 1 GB e processador de 1.3 GHz. O disco rígido está particionado com 1 GB para memória swap, 10 GB para o diretório raiz e o restante para armazenar as máquinas virtuais.

²Software livre, segundo a definição criada pela Free Software Foundation é qualquer programa de computador que pode ser usado, copiado, estudado, modificado e redistribuído sem nenhuma restrição.

Foi adicionada uma linha de comando no arquivo `/etc/profile` do usuário “aluno”, para que o mesmo possa usar as ferramentas descritas acima, sem ter que digitar todo o caminho, por exemplo, `/usr/sbin/traceroute`, mas apenas o comando `traceroute`. No arquivo `/etc/sudoers` foi adicionada a linha:

```
%aluno ALL=/usr/sbin/wireshark-root
```

para que o usuário “aluno” tenha acesso a ferramenta Wireshark como super-usuário.

Script criado para dar privilégios de super-usuário ao usuário “aluno”.

```
#!/bin/bash
cd /usr/sbin
chmod 7755 traceroute
chmod 7755 tcpdump

cd /sbin
chmod 7755 ifconfig
chmod 7755 ip
chmod 7755 arp

cd /etc/init.d
chmod 7755 sshd
```

Alteração no arquivo `/etc/profile`, do usuário “aluno”:

```
export PATH=$PATH:/usr/sbin:/sbin
```

4.2.1 Teoria e Prática no uso de Máquinas Virtuais

Máquinas Virtuais

Uma máquina virtual (*virtual machine VM*) é definida em (POPEK; GOLDBERG, 1974) como uma duplicata eficiente e isolada de uma máquina real. Um ambiente de máquina virtual é criado por um monitor de máquina virtual, também chamado de um “sistema operacional para sistemas operacionais” (KELEM; FEIERTAG, 1991). O monitor cria uma ou mais máquinas virtuais em uma única máquina real, fornecendo uma interface (através da multiplexação do hardware) que é idêntica ao hardware subjacente, controlando uma ou mais VMs. Cada VM provê facilidades para uma aplicação ou um “sistema convidado” que acredita estar sendo executado sobre um ambiente convencional com acesso direto ao hardware. Um monitor VM constrói algumas propriedades que são úteis no sistema de segurança como:

- Isolamento: Um software em execução numa VM não pode acessar ou modificar outro software em execução no monitor ou em outra VM;
- Inspeção: O monitor tem acesso e controle sobre todas as informações do estado da VM, como estado da CPU, conteúdo de memória, eventos, e assim por diante;
- Interposição: O monitor pode interceptar e modificar ações emitidas por uma VM;
- Gerenciabilidade: Como cada VM é uma entidade independente das demais, a administração das diversas instâncias é simplificada e centralizada.

O uso típico de sistemas de máquinas virtuais abrange o desenvolvimento e testes de novos sistemas operacionais e o funcionamento simultâneo de sistemas operacionais diferentes no mesmo hardware, denominado “consolidação de servidores”.

4.2.2 Tipos e Modelos de Máquinas Virtuais

Tipos de Máquinas Virtuais

Existem duas abordagens clássicas para a construção de sistemas de máquinas virtuais:

- Tipo I - O monitor de máquina virtual é implementado entre o hardware e o(s) sistema(s) convidados(s). Cada VM se comporta como uma máquina física completa que pode executar o seu próprio sistema operacional, semelhante a um sistema operacional tradicional que está no controle da máquina, como mostrado na figura 16. Os ambientes virtuais XEN e VMWare ESX Server são exemplos de sistemas do tipo I.
- Tipo II - O monitor é implementado como um processo normal de um sistema operacional real, denominado sistema anfitrião. O monitor de Tipo II funciona de forma análoga ao de Tipo I, sendo a sua maior diferença a existência de um sistema abaixo deste. Neste modelo, o monitor simula todas as operações que o sistema anfitrião controlaria, como mostrado na figura 17. O VMware Server e User Mode Linux são exemplos de sistemas do tipo II.

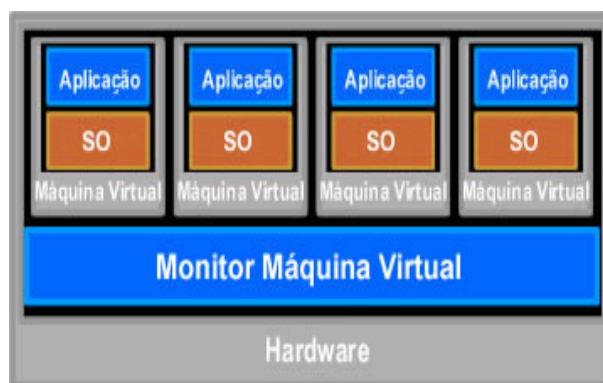


Figura 16: Monitor Tipo I

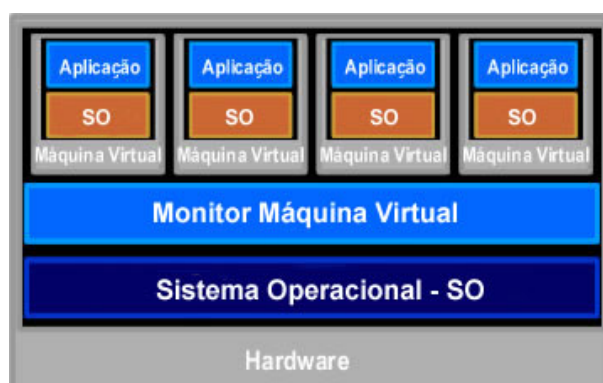


Figura 17: Monitor Tipo II

Modelos de Máquinas Virtuais

Existem alguns modelos de máquinas virtuais disponíveis para aplicações definidas. Segue quatro (04) modelos conhecidos:

User Mode Linux (UML) - O User-Mode Linux é uma VM de Tipo II, ou seja, executa na forma de um processo no sistema anfitrião, e os processos em execução na VM não têm acesso aos recursos do sistema anfitrião diretamente, sendo uma alternativa de uso de máquinas virtuais no ambiente Linux. O *kernel* do Linux é implementado de tal forma a suportar sobre si mesmo outro *kernel* do Linux. O armazenamento em disco para a máquina virtual é inteiramente contida dentro de um único arquivo em sua máquina física, separado e isolado na forma de uma VM que utiliza a simulação de hardware construída a partir dos serviços providos pelo sistema anfitrião. Essa VM é capaz de executar todos os serviços e aplicações disponíveis para o sistema anfitrião. Ou seja, se obtém um outro sistema operacional Linux, totalmente isolado do sistema que roda na máquina real; uma cópia da VM pode ser restaurada toda vez que for necessário sem prejuízo para o sistema anfitrião. Pode-se instalar várias VMs e simular uma rede, por exemplo, em uma única máquina. O UML roda em modo texto e contempla distribuições linux como Debian, Slackware, Mandriva e Suse.

VMware - Roda tanto no SO Linux como no SO Windows. Depois de instalado, permite a configuração de dispositivos como portas serial, paralela e usb, placa de rede, memória, tamanho de HD e outros dispositivos que serão usados pelo SO convidado. O armazenamento da máquina virtual na máquina física é feito em um único arquivo para cada instalação, podendo ser restaurado rapidamente em caso de danos no mesmo. A gerência de memória no VMware é feita diretamente pelo sistema convidado. Para garantir que não ocorra nenhuma colisão de memória entre o sistema convidado e o real, o VMware aloca uma parte da memória para uso exclusivo, então o sistema convidado utiliza essa memória previamente alocada.

Denali - A VM Denali executa diretamente no hardware sem necessidade de um sistema operacional anfitrião (ou seja, usa uma arquitetura de tipo I). Conseqüentemente, o monitor do Denali tem que prover drivers de dispositivo para todo o hardware da plataforma adotada. Pela implementação dos drivers de dispositivo no monitor, o Denali força políticas de multiplexação total do hardware. Isto garante o isolamento entre diferentes instâncias de máquinas virtuais, mas dificulta a implementação do monitor. Não existe o conceito de memória virtual no ambiente Denali. O sistema convidado é executado em um único espaço de memória privado. Esta abordagem simplifica o monitor, mas a falta de proteção de memória dentro da VM limita a capacidade do sistema convidado. Esta situação impõe alterações complexas no projeto de como as aplicações devem ser

construídas. A manipulação de interrupções também é diferenciada. Em vez de tratar interrupções quando elas acontecem, elas são colocadas numa fila até que a VM as execute. Isto reduz o número de interrupções que o monitor necessita tratar, e conseqüentemente, o custo de virtualização torna-se menor.

Xen - O ambiente Xen é um monitor de Tipo I para a plataforma x86. Suporta múltiplos sistemas convidados simultaneamente com bom desempenho e isolamento. A proposta do ambiente Xen é suportar aplicações sem a necessidade de alterações, múltiplos sistemas operacionais convidados e a cooperação entre estes sistemas, mas com o máximo de desempenho possível. O monitor Xen se encontra em um acentuado grau de maturidade e pode ser utilizado em sistemas de produção; o seu código fonte está liberado sob a licença GNU *General Public Licence* (GPL)³. Atualmente, o ambiente Xen suporta os sistemas Windows XP, Linux e Unix (baseado no NetBSD).

4.2.3 Uso de Máquinas Virtuais no Ensino de Redes de Computadores

No ensino de Redes de Computadores, além de uma base teórica sólida, é necessário o ensino prático, em laboratório, no qual o estudante constrói, configura e testa cenários de redes de computadores. Os recentes avanços na tecnologia de máquinas virtuais têm possibilitado o crescente uso destas em disciplinas relacionadas à área de redes de computadores, adotando-as como uma ferramenta importante no processo de aprendizagem.

Soma-se a este evento, a popularização de computadores domésticos com capacidade suficiente para executar várias máquinas virtuais simultaneamente, o que permite ao estudante reproduzir facilmente, em casa, experimentos feitos em classe, participando, desta forma, de todas as fases da construção, configuração e testes de cenários de redes de computadores.

³GNU General Public License (Licença Pública Geral) é a designação da licença para software livre idealizada por Richard Stallman no final da década de 1980, no âmbito do projecto GNU da Free Software Foundation (FSF). O projeto GNU foi iniciado em 1984 para desenvolver um Sistema Operacional completo, compatível com o UNIX, mas que fosse Livre: O sistema GNU. GNU é um acrônimo recursivo que significa GNU is not Unix (GNU não é Unix)!
fonte:<http://www.gnu.org/>

Alguns Benefícios do Uso de Máquinas Virtuais (NABHEN; MAZIERO, 2006):

- É possível criar mais *hosts* virtuais do que o número de máquinas físicas disponíveis em laboratório, permitindo que cada aluno crie cenários complexos envolvendo vários *hosts*;
- O número de interfaces de rede em cada *host* e sua interligação, são definidas no contexto virtual, sem nenhuma alteração na estrutura física do laboratório ou configuração de hardware do computador;
- O aluno é o administrador de seu *host* virtual, permitindo a mudança de sua configuração e instalação de softwares exigidos por cada experiência;
- O aluno pode salvar em sua máquina real as configurações e estado de cada *host* virtual, permitindo que se desenvolvam experiências mais complexas e incrementadas;

4.2.4 Uso de Máquinas Virtuais no Laboratório Integrado de Redes de Computadores

Antes das mudanças na estrutura física e lógica do laboratório de Redes de Computadores I, cada computador tinha seu disco rígido particionado para dois sistemas operacionais, conforme necessidade das disciplinas ministradas, descritas anteriormente. Alunos e professores possuíam senha de super-usuário, podendo fazer alterações (configurações de rede, por exemplo) que, na maioria das vezes, causavam danos ao sistema operacional ou tornava-o inoperante para uso em outra disciplina. Em razão deste procedimento, os computadores freqüentemente estavam em manutenção, diminuindo consideravelmente a quantidade de máquinas disponíveis para uso no laboratório. Decorre do exposto a idéia de instalar máquinas virtuais nos computadores; estas permitiriam o uso de sistemas operacionais diferentes e que poderiam ser reinstalados rapidamente, sem necessidade de recolher as máquinas à manutenção. Atualmente, cada computador do laboratório utiliza no mínimo duas máquinas virtuais instaladas, uma rodando o SO Mandriva Linux 2008 e outra o SO Windows XP. Havendo algum problema com estes SOs convidados, eles simplesmente são copiados, em poucos minutos, de um outro computador via rede ou de um DVD-ROM, evitando-se quaisquer transtornos.

4.2.5 Modelo de Máquina Virtual usada no Laboratório de Redes de Computadores I

Foi usado a máquina virtual VMware Server 1.0.4. Sua escolha se deu pela exigência de hardware ser compatível com a configuração de hardware dos computadores do laboratório, pela facilidade na instalação e configuração, por apresentar interface gráfica para os SOs convidados e por ser esta versão distribuída gratuitamente ⁴. Os detalhes de instalação e configuração estão descritos nos anexos 7.1 e 7.2.

O SO convidado Windows XP é utilizado para rodar software proprietário de configuração de *switchs* e roteadores, bem como o ambiente DOS para software do minicontrolador 8051.

No SO convidado Mandriva Linux 2008 pode-se alterar todas as configurações necessárias para o bom desempenho das disciplinas ministradas bem como usar todas as ferramentas disponíveis como super-usuário, pois a senha de super-usuário é liberada para todos os usuários. Dependendo do tamanho da cada máquina virtual e do espaço livre no disco rígido, podem-se criar vários cenários virtuais de redes de computadores em um único computador, podendo-se usufruir de todas as vantagens descritas anteriormente. São exemplos de cenários de redes de computadores que podem ser criados em um computador real:

- Criação de uma VPN;
- Testes de serviços (cliente - servidor) como o SSH, FTP, entre outros;
- Teste de segurança (firewall);
- Criação de redes e configuração de roteadores.

Usando outros computadores podem ser criados os seguintes cenários:

- Criação de sub-redes e configuração de roteadores;

⁴O download do software é feito no sítio www.vmware.com/downloads/server, sendo escolhido o formato rpm pela facilidade de instalação.

- Captura de pacotes, identificar rotas em uma única rede virtual;
- Usar várias tecnologias de redes (sem fio, Ethernet) em máquinas virtuais sem alterar em nada a estrutura do laboratório.

5 Cenários de Redes de Computadores

5.1 Cenários de Redes Usando os Equipamentos do Laboratório Proposto

Nesta sessão são descritos três exemplos de aplicações realizadas no laboratório integrado e flexível de redes de computadores.

No primeiro exemplo, ilustrado na figura 18, foi realizado uma comunicação de dados através de uma linha discada entre pares de computadores, utilizando modems analógicos e uma central telefônica PABX. Com esta estrutura, os alunos poderão explorar os recursos do modems analógicos como os comandos AT, montar as conexões físicas, bem como realizar a configuração dos equipamentos e dos software de comunicação envolvidos (GALLO; HANCOCK, 2003).

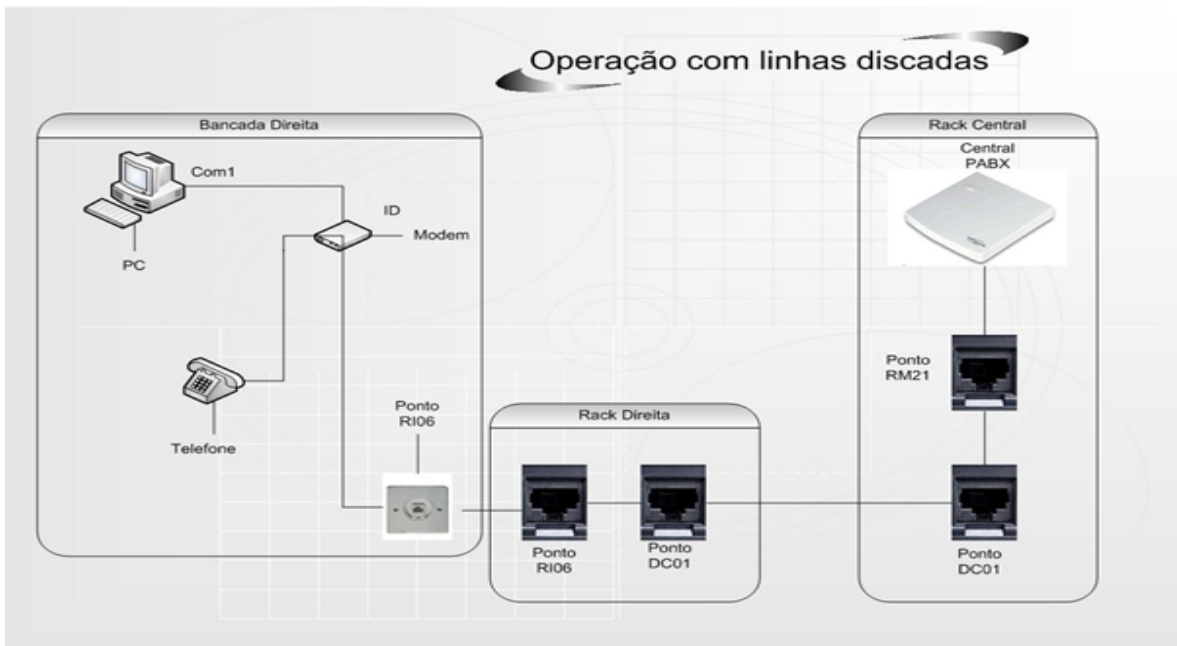


Figura 18: Aplicação com Linha Discada.

A descrição completa deste exemplo, bem como os testes realizados são apresentados no anexo 7.3.

No segundo exemplo, ilustrado na figura 19, foi criada três redes distintas ligadas entre si por três roteadores e modems. Este exemplo foi criado usando a estrutura física do laboratório de Redes I, porém, com o uso de *hosts* virtuais. Alcançou-se os objetivos esperados, sendo que a descrição completa deste experimento está no anexo 7.4.

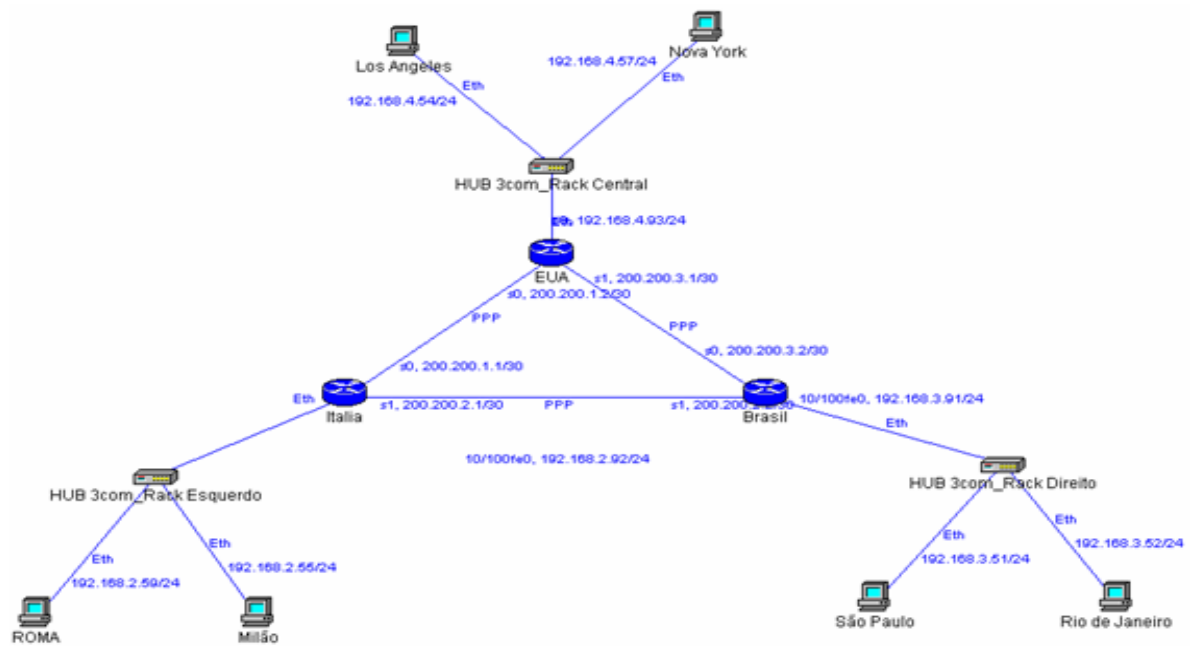


Figura 19: Estrutura Redes de Roteadores.

No terceiro exemplo, ilustrado na figura 20, foi feita a configuração de dois roteadores interligando duas sub-redes, realizada para teste de protocolos de roteamento. Sobre esta estrutura serão trabalhados os conceitos de sub-redes, com os alunos configurando as interfaces de rede nas máquinas virtuais e realizando testes de conectividade com `ping`. Serão configurados os roteadores, utilizando a console e também acesso remoto com TELNET, estabelecendo tabelas estáticas de roteamento e realizados testes de conectividade entre as máquinas de diferentes sub-redes, traçado de rotas com `traceroute`. Finalmente, serão instalados protocolos de roteamento nos roteadores (RIP e OSPF), verificando o aprendizado das rotas pelos roteadores. A descrição completa deste exemplo, bem como os testes realizados são apresentados no anexo 7.5.

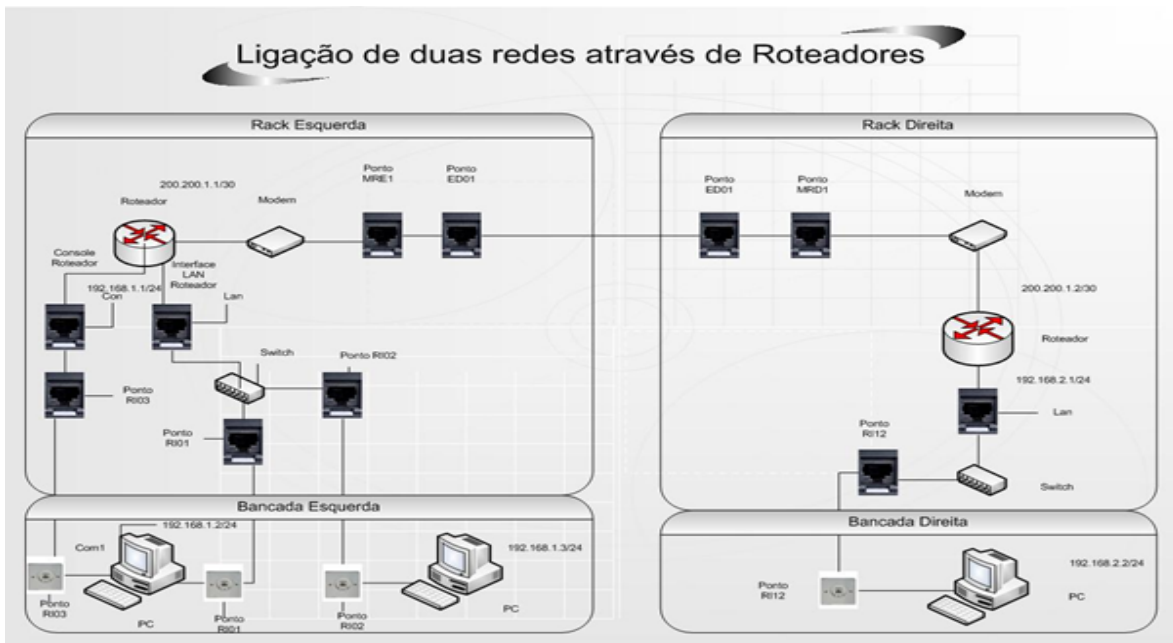


Figura 20: Estrutura Interligando Duas Redes de Roteadores.

5.2 Cenários de Redes com Máquinas Virtuais

Nesta sessão são descritos três exemplos de aplicações realizadas usando a tecnologia de máquinas virtuais.

No primeiro exemplo foi configurado uma malha com três roteadores interligando clientes de três redes distintas como mostrado na figura 21.

Esta malha foi configurada usando apenas um computador real e instaladas neste 7 máquinas virtuais. Não houve nenhuma interferência de uma máquina virtual em outra, nem das máquinas virtuais na máquina real e nem da máquina real nas máquinas virtuais. Houve comunicação entre cada host virtual conforme esperado. Concluímos com esse exemplo que cada estudante poderá montar sua própria rede virtual e comprovamos a eficácia do uso de máquinas virtuais no estudo desta área. A descrição completa deste exemplo, bem como os testes realizados são apresentados no anexo 7.6.

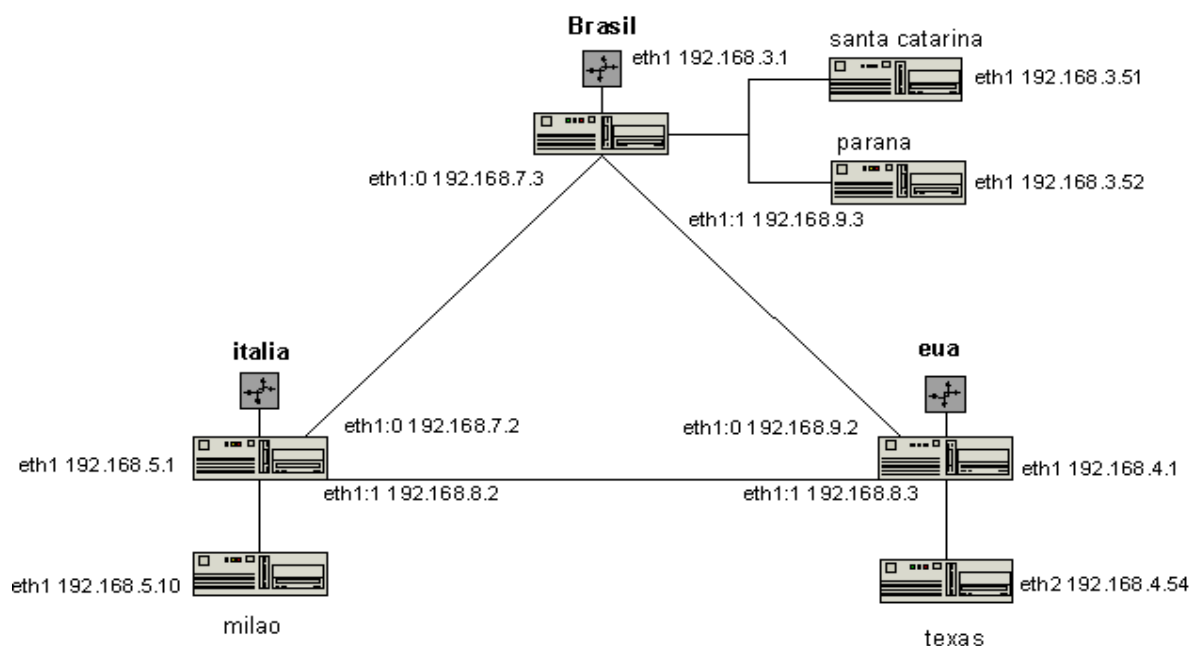


Figura 21: Diagrama de Redes de Roteadores.

No segundo exemplo foi configurada uma rede virtual com acesso à Internet no laboratório de redes I usando a infra-estrutura do mesmo. Não foi alterada a infra-estrutura nem as configurações da rede do laboratório como mostrado na figura 22.

Esta malha foi configurada utilizando uma máquina virtual em cada máquina real, sendo que destas máquinas, apenas a máquina do professor foi configurada como roteador, permitindo acesso a rede do CEFET/SC. Este exemplo permitiu-nos criar uma rede independente da rede real, configurá-la, alterá-la, incluir *hosts* e ao final do experimento, se for o caso, reinstalar a máquina virtual para uso de outra disciplina em poucos minutos. A descrição completa deste exemplo, bem como os testes realizados são apresentados no anexo 7.7.

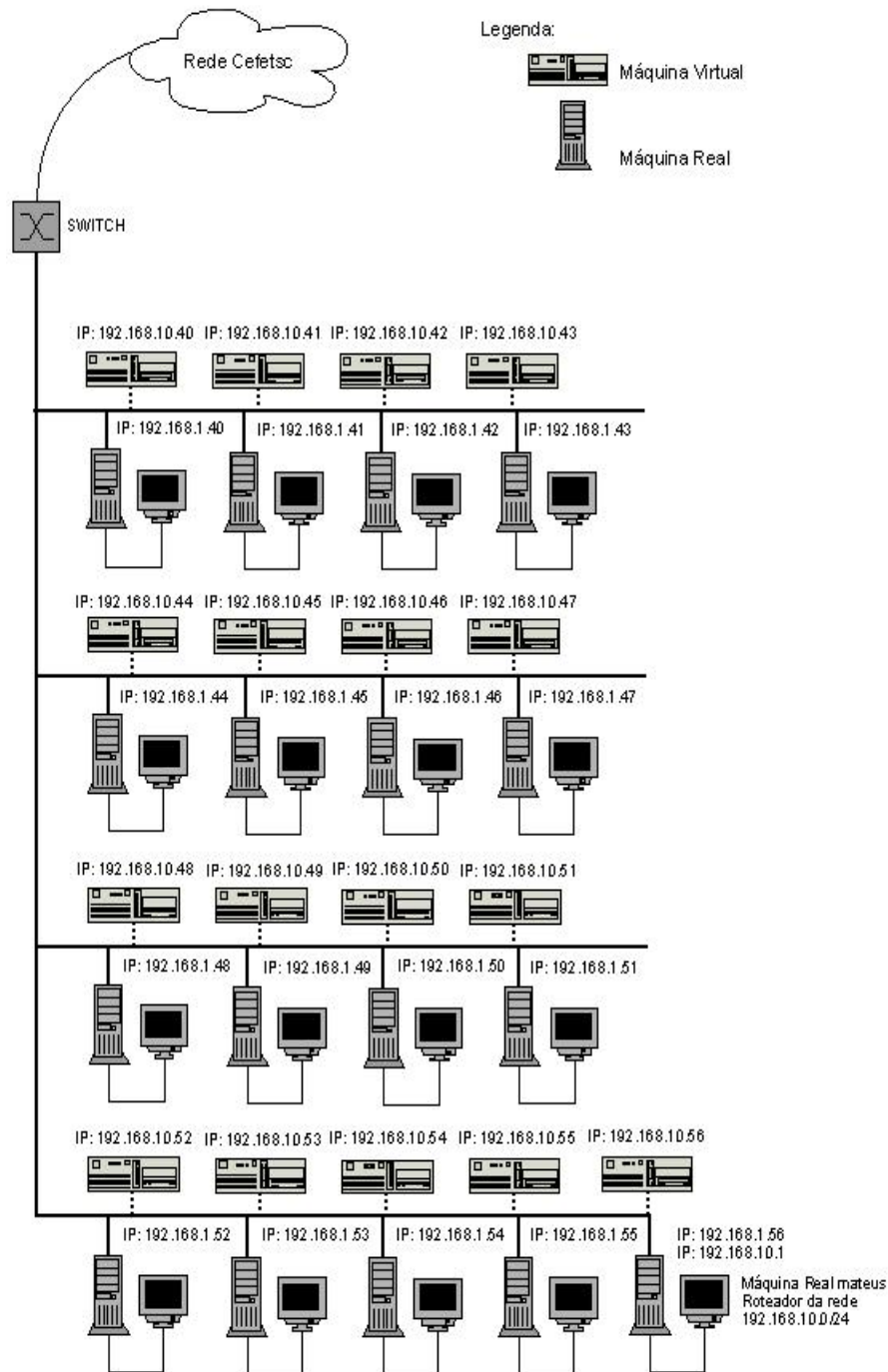


Figura 22: Diagrama de Rede Virtual com Acesso a Internet no Laboratório de Redes I.

No terceiro exemplo foi configurada uma malha de rede virtual com acesso à Internet no laboratório de redes I usando a infra-estrutura do mesmo como mostrado na figura 23. Não foi alterada a infra-estrutura nem as configurações da rede do laboratório. Esta malha foi configurada utilizando uma máquina virtual em cada máquina real. Oito destas máquinas foram configuradas como roteadores e oito como clientes. Todas as máquinas

ficaram conectadas entre si e a máquina do professor com acesso a rede do CEFET/SC. Este exemplo permitiu-nos criar várias redes independentes configurá-las, alterá-las, incluir *hosts* e todas independentes da rede real do laboratório de Redes I. A descrição completa deste exemplo, bem como os testes realizados são apresentados no anexo 7.8.

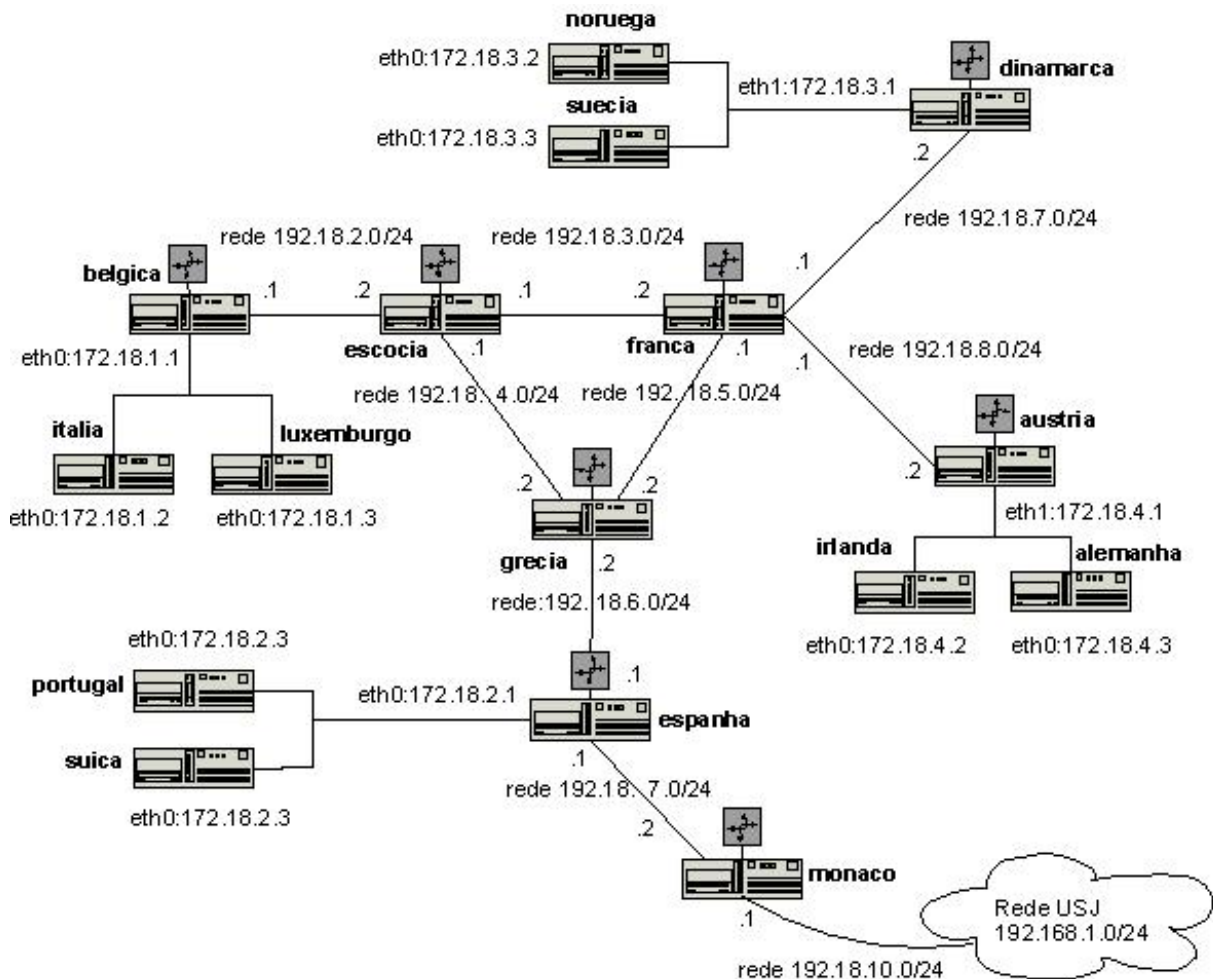


Figura 23: Diagrama da Malha de Rede Virtual com Acesso a Internet no Laboratório de Redes I.

6 Conclusões

Ao iniciarmos o projeto do Laboratório de Redes Integrado sabíamos da importância de um laboratório estruturado para o ensino de redes de computadores. Em nossas aulas no CST em Sistemas de Telecomunicações, nas disciplinas de Redes de Computadores II e III, havíamos passado por diversas dificuldades na montagem de experimentos. Como citado no capítulo 4, os equipamentos ficavam guardados em armários, pois não havia um local específico para ficarem fixados. As montagens eram feitas sobre as bancadas dos computadores, dividindo espaço com os mesmos, tendo um emaranhado de cabos e ferramentas para conexão. Perdíamos muito tempo na montagem de um experimento, às vezes uma ou duas aulas.

Diante desta realidade, pesquisamos sobre o ensino-aprendizagem de redes de computadores, reunindo no capítulo 2 a opinião de vários autores. Pode-se perceber que os mais diversos modelos de cursos apresentados possuem certa semelhança, sempre passando pelos tópicos de comunicação entre redes, programação voltada à redes de computadores, manuseio de equipamentos como hubs, switches, modems e roteadores. Este estudo permitiu-nos uma visão mais abrangente sobre a importância das atividades práticas no ensino de redes de computadores.

A necessidade de aliar o ensino prático ao ensino teórico é fundamental, como descrito por Kurose et al. (2002). Dificilmente o estudante terá uma visão abrangente do que é redes de computadores tendo apenas o ensino teórico, que, segundo Kurose (2002) “atrela ao ensino de redes de computadores um certo ar de abstração”. Por isso mesmo, fazem referência à necessidade da elaboração de laboratórios exclusivos para o ensino de redes de computadores, laboratórios estes que forneçam os recursos exigidos para a realização das mais diversas experiências.

Os exemplos de laboratórios, citados no capítulo 3, serviram de base para que pudéssemos desenvolver nosso projeto, o qual foi descrito no capítulo 4, aliado as necessidades dos professores da área de redes de computadores. Outro tema importante foi o uso da tecnologia de máquinas virtuais. Como exposto no capítulo 4, o objetivo principal era resolver os problemas de mal funcionamento das máquinas após as aulas de laboratório. Porém, ao estudarmos mais sobre esta tecnologia, pudemos perceber o quanto seria importante seu uso no ensino de redes de computadores. Exemplos de experimentos foram descritos no capítulo 5, sendo que os detalhes da configuração de cada experimento e seus resultados foram apresentados nas sessões 7.3, 7.4, 7.5, 7.6, 7.7 e 7.8.

Uma proposta de continuidade dos trabalhos no laboratório integrado de redes de computadores é o desenvolvimento de estruturas híbridas, envolvendo equipamentos físicos e máquinas virtuais, visando a análise do funcionamento de protocolos de roteamento intra-domínio e inter-domínio.

O laboratório integrado de redes de computadores vem sendo utilizado de maneira sistemática pelos professores e alunos de redes de computadores do CEFET/SC, além de ser compartilhado com outras unidades curriculares. A estrutura flexível do laboratório tem permitido a realização de experimentos diversos, facilitando enormemente as montagens e a visualização das estruturas físicas pelos alunos. Também a parte lógica do laboratório, utilizando máquinas virtuais, reduziu os problemas de mal funcionamento das máquinas, antes muito comuns após as aulas de laboratório. Assim como os modelos de laboratório apresentados em (KUROSE et al., 2002), acreditamos que o laboratório integrado e flexível aqui apresentado, pode contribuir com o aprimoramento do ensino-aprendizagem de redes de computadores, dando suporte a um de seus aspectos fundamentais, que é a realização de atividades práticas.

7 *Anexos*

7.1 Instalação do VMware Server 1.0.4

- Baixe o arquivo de instalação do VMware, compatível com seu sistema operacional no sítio <http://www.vmware.com/download/server/>.
- Instale o kernel-source (que deve ter a versão compatível com a versão Linux em uso) e o `xinetd` com os comandos:
 - `urpmi kernel-source` (kernel-devel no Mandriva 2008)
 - `urpmi xinetd`
- Se foi baixada a versão tar.gz desempacotar o `vmware-server` usando:
 - `tar -zxvf VMware-server-1.0.xxx...`
 - Na pasta criada: **vmware server distrib**, digitar `./vmware-install.pl`
 - Confirme todas as perguntas com o default apresentado e diga yes para a licença. Continue confirmado os defaults até colocar o número serial das licenças que foram fornecidas pela VMware no ato do cadastro e baixa dos arquivos pelo site. A instalação deste aplicativo é encerrada.
- Retorne para o diretório **vmware** e desempacote o **vmware-server-console** usando:
 - `tar -zxvf VMware-server-consolexxx...`
 - Configure o mesmo com o comando: `/usr/bin/vmware-config.pl`
 - Confirme todas as perguntas com o default apresentado e diga yes para a licença. Continue confirmado os defaults até colocar o número serial das licenças

que foram fornecidas pela VMware no ato do cadastro e download dos arquivos pelo sítio. A instalação deste aplicativo é encerrada.

- Se for baixada a versão .rpm :
 - urpmi **nome do arquivo**
 - Digite `/usr/ bin/vmware-config.pl`
 - Confirme todas as perguntas com o default apresentado e digite **yes** para a licença. Continue confirmado os defaults até colocar o número serial das licenças que foram fornecidas pela VMware no ato do cadastro e download dos arquivos pelo sítio. A instalação deste aplicativo é encerrada.

7.2 Configuração do VMware Server 1.0.4

1. Configurada a aplicação na máquina real, no **menu iniciar** do Mandriva Linux 2008, selecione **executar programa**, digite *vmware* e tecle **enter**; ou no terminal do Mandriva Linux 2008 (modo texto) digitar *vmware* (não precisa ser **root** e nem estar em diretório específico).
2. Utilize a opção **localhost** na abertura;
3. Use a opção para criar uma nova máquina virtual seguindo os passos do *wizard*:
 - (a) Selecione configuração típica ;
 - (b) Escolha o sistema operacional que será instalado (Mandriva Linux 2008);
 - (c) Use o caminho sugerido pelo VMware para alocação da instalação da máquina virtual ou use uma partição criada para este fim;
 - (d) Selecione *Network Connection* e use *Bridge Networking*;
 - (e) No *disk Size*, coloque o valor da partição para 5 GB . Mantenha selecionadas as duas opções abaixo (alocar todo o espaço e *split disk*);
 - (f) Se não existir erro nas configurações anteriores é só aguardar a criação do disco.
4. Criada a máquina virtual, no lado direito do aplicativo será apresentado os itens de *hardware* padrão para cada máquina criada. Adicionar neste momento novos itens de hardware antes de instalar o sistema operacional como, serial, USB, porta

paralela e áudio. Usar a opção *edit virtual machine settings*. Ajustar a memória, verificando as dicas de tamanho mínimo, recomendável e tamanho máximo ou digite um tamanho de memória que deve ser a metade ou inferior ao que está na máquina;

5. Inicie com *Power ON* a máquina virtual. Uma tela do VMware aparece seguida do *boot* no CD-ROM. Coloque o CD de instalação do Mandriva Linux 2008 e siga a instalação como normalmente se faz. Para que a leitura do CD ocorra clique na tela da máquina virtual (no aplicativo) e após **enter**.
6. Pode-se usar a tela cheia usando a opção *full screen* no menu superior no aplicativo. Para sair do ambiente virtual em qualquer momento, é só clicar *ctrl-alt*.

7.3 Comunicação entre Computadores Utilizando Linha Discada

Este exemplo, ilustrado na figura 24, permite realizar uma comunicação de dados através de uma linha discada entre pares de computadores, utilizando modems analógicos e uma central telefônica PABX. O PABX está instalado no Rack Central do laboratório e tem seus ramais acessíveis diretamente em um *patch panel*, com identificação própria (p. ex. RM21). O par computador/modem, mostrado na figura 24, está situado junto ao ponto da Rede Interna RI06. Para realizar a conexão do modem com o PABX, no Rack Direito, conectou-se o ponto de RI06 com o ponto DC01, que é um ponto de espelhamento do Rack Direito no Rack Central. No Rack Central, por sua vez, conectou-se o ponto de espelhamento DC01 com o ponto RM21, completando-se a conexão. De forma análoga, outros pares computador/modem foram conectados a outros ramais do PABX para viabilizar a comunicação de dados entre pares de computadores. Com esta estrutura, os alunos

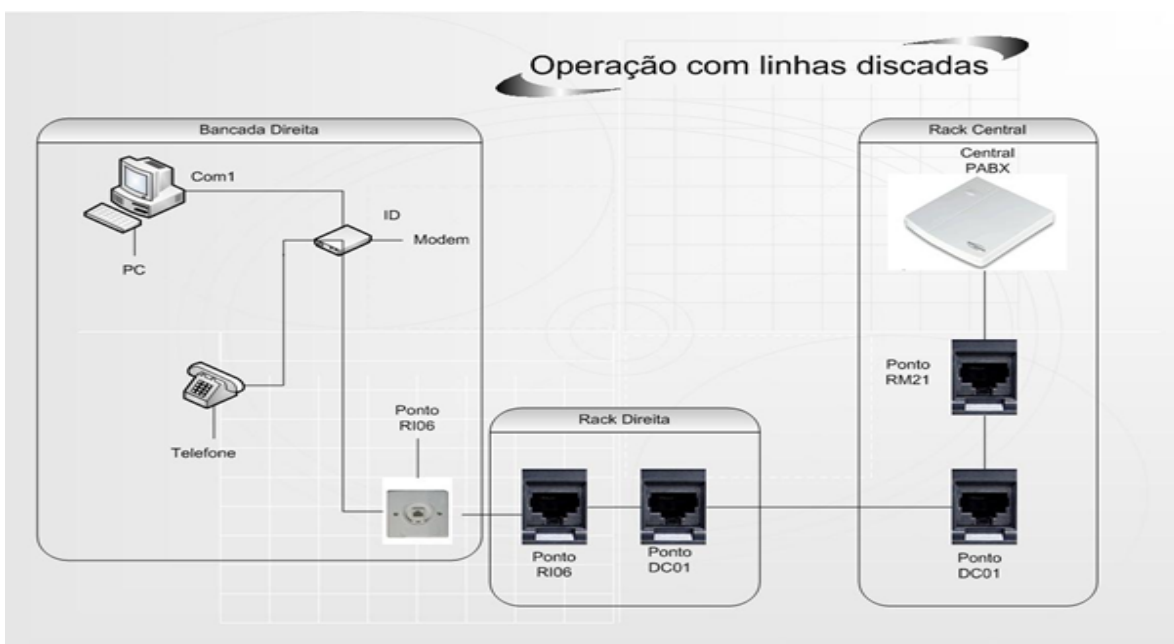


Figura 24: Aplicação com Linha Discada.

puderam explorar os recursos dos modems analógicos, como os comandos AT, montar as conexões físicas, bem como realizar a configuração dos equipamentos e dos software de comunicação envolvidos (GALLO; HANCOCK, 2003).

7.4 Instalação e Configuração da Rede de Roteadores

Para a realização do referido experimento, primeiramente, fez-se necessária a configuração das máquinas que seriam utilizadas para tal. Desta forma, como primeiro passo foi inicializar a máquina virtual correspondente ao sistema operacional Linux Mandriva. A utilização de máquina virtual nesse experimento ressalta a importância da sua utilização no laboratório, uma vez que o sistema operacional real do host não foi utilizado diretamente e, por isso, não foram alteradas as suas configurações. Com a máquina virtual já devidamente inicializada e pronta para ser usada, foi feita, portanto, a configuração de sua interface de rede. Para isso, foi editado o arquivo `ifcfg-ethX`, onde X corresponde ao número da interface que está sendo configurada. Para editar esse arquivo, foi feito o seguinte comando:

```
vi /etc/sysconfig/network-scripts/ifcfg-ethX.
```

Observa-se abaixo respectivo arquivo editado, tomando como exemplo o *host* 192.168.1.2/24. Neste caso foi atribuído o IP 192.168.1.2, com máscara de rede 255.255.255.0 e *gateway* 192.168.1.1.

- `BOOTPROTO = static;`
- `IPADDR = 192.168.1.2;`
- `NETMASK = 255.255.255.0.`
- `GATEWAY = 198.168.1.1`

Após este procedimento, foi reiniciado o serviço de rede do sistema operacional, através do seguinte comando: `service network restart` Assim estabeleceu-se na máquina o sua configuração de rede para o ambiente desejado. Através do comando `ifconfig`, é possível observar a configuração de rede atual da máquina.

```
eth0      Link encap:Ethernet Endereço de HW 00:0C:29:05:9B:1C
          inet end.: 192.168.1.2 Bcast:192.168.1.255  Masc:255.255.255.0
          endereço inet6: fe80::20c:29ff:fe05:9b1c/64 Escopo:Link
          UP BROADCASTRUNNING MULTICAST  MTU:1500  Métrica:1
          RX packets:466 errors:0 dropped:0 overruns:0 frame:0
```

```
TX packets:257 errors:0 dropped:0 overruns:0 carrier:0
colisões:0 txqueuelen:1000
RX bytes:74146 (72.4 KiB) TX bytes:24759 (24.1 KiB)
IRQ:17 Endereço de E/S:0x1400
```

```
lo      Link encap:Loopback Local
        inet end.: 127.0.0.1 Masc:255.0.0.0
        endereço inet6: ::1/128 Escopo:Máquina
        UP LOOPBACKRUNNING MTU:16436 Métrica:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        colisões:0 txqueuelen:0
        RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
```

Este procedimento foi realizado com cada máquina de acordo com sua respectiva rede e, conseqüentemente, seu respectivo endereço e *gateway*. Estando as máquinas devidamente configuradas e se comunicando em suas sub-redes, restou a configuração dos dois roteadores. Foram utilizados os roteadores **Cisco 1750**. A sua configuração se dá através da ligação da sua porta de console a uma das tomadas de bancada através de cordões de manobra conectados nos respectivos *racks*. Através então do aplicativo **Hyper Terminal**, fez-se a comunicação com a console do roteador através da porta *Com1* do computador conectada à tomada que estava em comunicação com a porta *console* do roteador. Seguiu-se, então os seguintes comandos:

```
<enter> password: a
```

```
enable(para ter acesso a comandosprivilegiados) password: a
```

```
configure terminal (para ter acesso aos comandos de configuração)
```

```
interface serial 0
```

```
no shutdown
```

```
ip address 200.200.1.x 255.255.255.252
```

```
bandwidth 64
```

```
encapsulation ppp
```

```
interface fastethernet 0
```

```
no shutdown
```

```
ip address 192.168.y.z 255.255.255.0
```

Onde **x** é o último número que define o endereço do roteador em sua **WAN**, **y** é o número da rede da **LAN** e **z** o último número que define o endereço da **LAN** do roteador. Além destas configurações lógicas, foi necessária também a configuração dos modems que fariam a ligação entre as **WAN's** dos roteadores. Da mesma maneira, é possível aumentar a dimensão deste experimento utilizando um roteador e um *hub* do rack central, como observa-se na figura 25 a seguir:

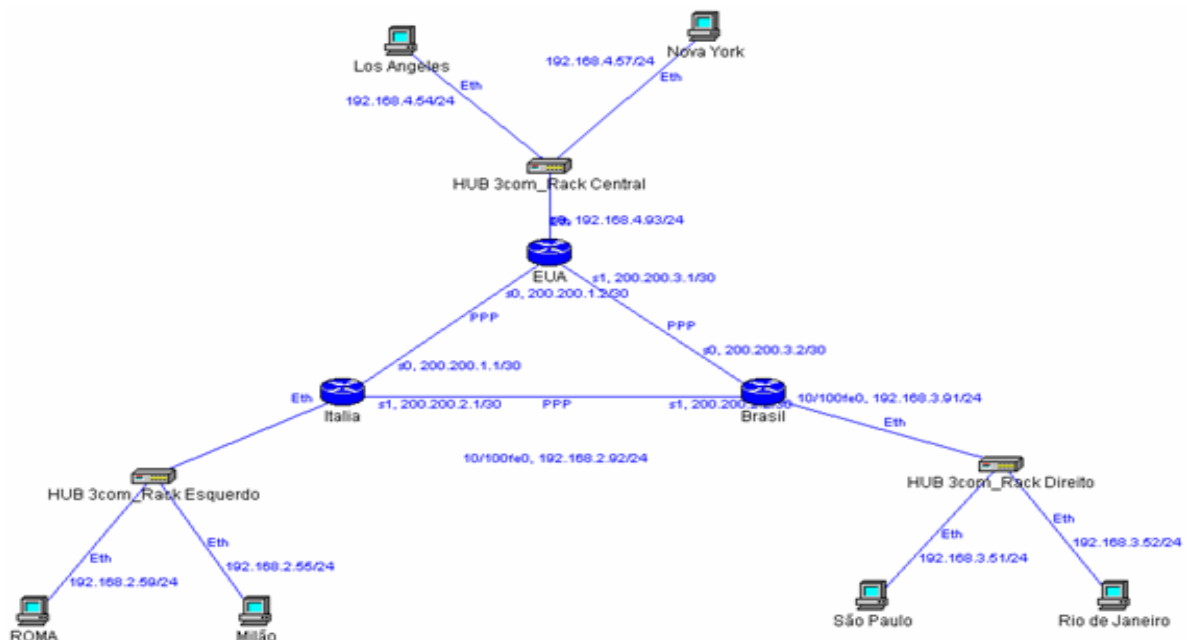


Figura 25: Estrutura Redes de Roteadores.

Para implementar este experimento, basta aplicar as mesmas configurações que foram feitas no experimento anterior e aplica-las nas demais máquinas e demais interfaces de rede. Observa-se agora uma ligeira diferença. Os roteadores necessitam de suas duas

interfaces **WAN**, portanto, sendo necessários dois modems para cada um deles.

7.4.1 Testes com PING

Comunicação entre o host 192.168.2.55 e o host 192.168.4.54:

```
PING 192.168.4.54 (192.168.4.54) 56(84) bytes of data. 64 bytes from
192.168.4.54: icmp_seq=1 ttl=62 time=44.9 ms 64 bytes from
192.168.4.54: icmp_seq=2 ttl=62 time=44.5 ms 64 bytes from
192.168.4.54: icmp_seq=3 ttl=62 time=44.8 ms 64 bytes from
192.168.4.54: icmp_seq=4 ttl=62 time=44.7 ms 64 bytes
```

Comunicação entre o host 192.168.4.57 e o host 192.168.3.52:

```
PING 192.168.3.52 (192.168.3.52) 56(84) bytes of data. 64 bytes from
192.168.3.52: icmp_seq=1 ttl=62 time=46.2 ms 64 bytes from
192.168.3.52: icmp_seq=2 ttl=62 time=44.8 ms 64 bytes from
192.168.3.52: icmp_seq=3 ttl=62 time=45.9 ms 64 bytes from
192.168.3.52: icmp_seq=4 ttl=62 time=44.6 ms 64 bytes
```

Comunicação entre o host 192.168.3.51 e o host 192.168.2.59:

```
PING 192.168.2.59 (192.168.2.59) 56(84) bytes of data. 64 bytes from
192.168.2.59: icmp_seq=1 ttl=62 time=49.2 ms 64 bytes from
192.168.2.59: icmp_seq=2 ttl=62 time=44.3 ms 64 bytes from
192.168.2.59: icmp_seq=3 ttl=62 time=44.9 ms 64 bytes from
192.168.2.59: icmp_seq=4 ttl=62 time=44.1 ms 64 bytes
```

7.4.2 Testes com Traceroute

do host 192.168.2.55 até o host 192.168.4.54 da outra subrede, passando pelo seu gateway e também pelo roteador da rede remota:

1 192.168.2.92 (192.168.2.92) 1.266 ms 1.025 ms 1.021 ms

2 200.200.1.1 (200.200.1.1) 42.786 ms 44.326 ms 37.478 ms

3 192.168.4.54 (192.168.4.54) 39.587 ms 35.888 ms 36.678 ms

do host 192.168.4.57 até o host 192.168.3.52 da outra subrede, passando pelo seu gateway e também pelo roteador da rede remota:

1 192.168.4.93 (192.168.4.93) 1.765 ms 1.005 ms 1.981 ms

2 200.200.3.1 (200.200.3.1) 43.593 ms 44.876 ms 38.377 ms

3 192.168.3.52 (192.168.3.52) 36.667 ms 35.980 ms 39.032 ms

do host 192.168.3.51 até o host 192.168.2.59 da outra subrede, passando pelo seu gateway e também pelo roteador da rede remota:

1 192.168.3.91 (192.168.3.91) 1.450 ms 1.760 ms 1.009 ms

2 200.200.2.2 (200.200.2.2) 45.578 ms 45.076 ms 39.237 ms

3 192.168.2.59 (192.168.2.59) 38.867 ms 32.008 ms 37.980 ms

7.5 Teste de protocolos de roteamento em malha de dois roteadores

Este exemplo, ilustrado na figura 26, descreve a configuração de dois roteadores interligando duas sub-redes, realizada para teste de protocolos de roteamento. A conexão ponto-a-ponto entre os roteadores foi realizada por meio de modems digitais, simulando uma linha privativa de comunicação de dados (LPCD), estando cada par roteador/modem instalado em um *rack*. Para realizar a conexão entre os modems, no Rack Esquerdo

conectou-se o ponto de RME1 (interfaces RX e TX do modem da esquerda) com o ponto ED01, que é um ponto de espelhamento do Rack Esquerdo no Rack Direito. No Rack Direito, por sua vez, conectou-se o ponto de espelhamento ED01 com o ponto RMD01, completando-se a conexão. Para o roteador instalado no Rack da Esquerda, a interface

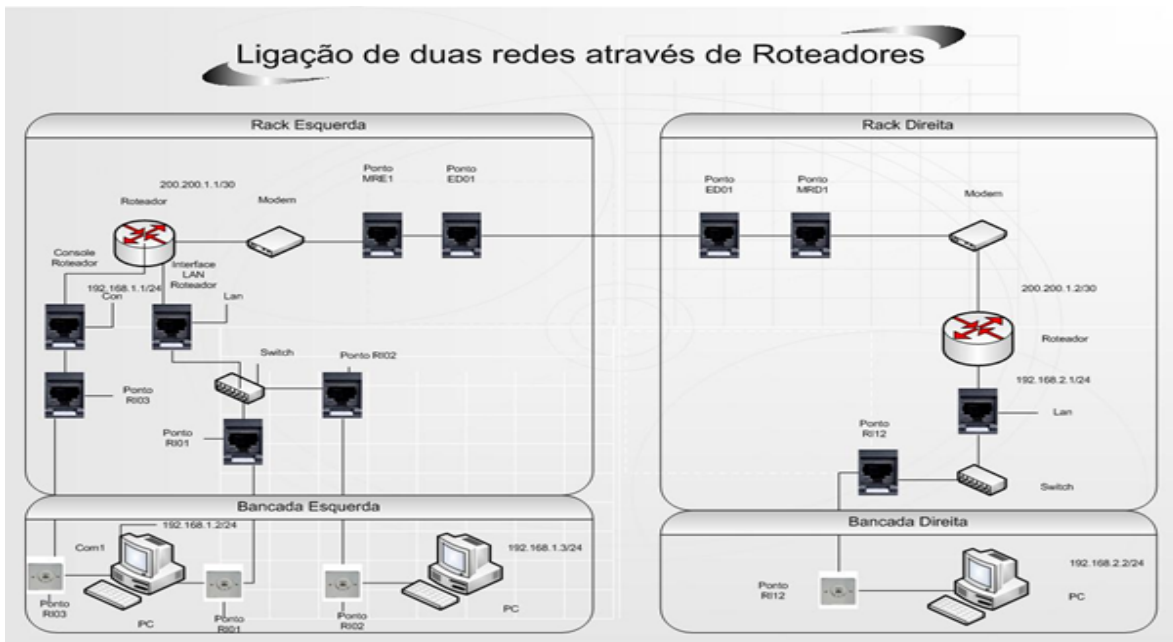


Figura 26: Estrutura Interligando Duas Redes de Roteadores.

LAN foi interligada a um switch e a dois computadores da bancada, situados junto aos pontos RI01 e RI02, formando a sub-rede 192.168.1.0/24. A console deste roteador foi conectada na porta serial de um dos computadores, situado junto ao ponto RI03. De maneira análoga, a interface LAN do outro roteador foi conectada a um switch e a um computador da bancada, formando a sub-rede 192.168.2.0/24.

Sobre esta estrutura foram trabalhados os conceitos de sub-redes, com os alunos configurando as interfaces de rede nas máquinas virtuais e realizando testes de conectividade com `ping`. Em seguida foram configurados os roteadores, utilizando a console e também acesso remoto com `TELNET`, estabelecendo tabelas estáticas de roteamento. Foram realizados testes de conectividade entre as máquinas de diferentes sub-redes e traçado de rotas com `traceroute`. Depois foram instalados protocolos de roteamento nos roteadores (`RIP` e `OSPF`), verificando o aprendizado das rotas pelos roteadores. Os testes realizados são apresentados abaixo.

7.5.1 Testes com PING

Para demonstrar a funcionalidade da experiência descrita na sessão 5.1, exemplo 2, lista-se a seguir os testes de comunicação realizados entre as interfaces de rede. Para esta demonstração utilizou-se um *host* de cada sub-rede.

HOST 192.168.1.2

Comunicação entre o host 192.168.1.2 e o host 192.168.1.3:

```
PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data. 64 bytes
from 192.168.1.3: icmp_seq=1 ttl=64 time=8.19 ms 64 bytes
from 192.168.1.3: icmp_seq=2 ttl=64 time=4.06 ms 64 bytes
from 192.168.1.3: icmp_seq=3 ttl=64 time=4.50 ms 64 bytes
from 192.168.1.3: icmp_seq=4 ttl=64 time=1.01 ms 64 bytes
from 192.168.1.3: icmp_seq=5 ttl=64 time=2.52 ms
```

```
--- 192.168.1.3 ping statistics ---
```

```
5 packets transmitted, 5 received, 0% packet loss, time 4002ms
rtt min/avg/max/mdev = 1.016/4.060/8.197/2.406 ms
```

Comunicação entre o host 192.168.1.2 e a interface ethernet do roteador:

```
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data. 64 bytes
from 192.168.1.1: icmp_seq=1 ttl=255 time=1.04 ms 64 bytes
from 192.168.1.1: icmp_seq=2 ttl=255 time=0.927 ms 64 bytes
from 192.168.1.1: icmp_seq=3 ttl=255 time=0.949 ms 64 bytes
from 192.168.1.1: icmp_seq=4 ttl=255 time=0.916 ms 64 bytes
from 192.168.1.1: icmp_seq=5 ttl=255 time=0.892 ms 64 bytes
from 192.168.1.1: icmp_seq=6 ttl=255 time=0.909 ms 64 bytes
from 192.168.1.1: icmp_seq=7 ttl=255 time=0.902 ms
```

```
--- 192.168.1.1 ping statistics ---
```

```
7 packets transmitted, 7 received, 0% packet loss, time 6001ms
```

rtt min/avg/max/mdev = 0.892/0.934/1.046/0.056 ms

Comunicação entre o host 192.168.1.2 e a interface serial 0 do roteador:

```
PING 200.200.1.1 (200.200.1.1) 56(84) bytes of data. 64 bytes
from 200.200.1.1: icmp_seq=1 ttl=255 time=1.57 ms 64 bytes
from 200.200.1.1: icmp_seq=2 ttl=255 time=0.945 ms 64 bytes
from 200.200.1.1: icmp_seq=3 ttl=255 time=1.24 ms 64 bytes
from 200.200.1.1: icmp_seq=4 ttl=255 time=1.15 ms 64 bytes
from 200.200.1.1: icmp_seq=5 ttl=255 time=1.02 ms 64 bytes
from 200.200.1.1: icmp_seq=6 ttl=255 time=1.00 ms 64 bytes
from 200.200.1.1: icmp_seq=7 ttl=255 time=0.989 ms 64 bytes
from 200.200.1.1: icmp_seq=8 ttl=255 time=1.15 ms
```

--- 200.200.1.1 ping statistics ---

8 packets transmitted, 8 received, 0% packet loss, time 7003ms

rtt min/avg/max/mdev = 0.945/1.137/1.576/0.195 ms

Comunicação entre o host 192.168.1.2 e a interface serial 0 do roteador remoto:

```
PING 200.200.1.2 (200.200.1.2) 56(84) bytes of data. 64 bytes
from 200.200.1.2: icmp_seq=1 ttl=254 time=44.6 ms 64 bytes
from 200.200.1.2: icmp_seq=2 ttl=254 time=44.7 ms 64 bytes
from 200.200.1.2: icmp_seq=3 ttl=254 time=44.5 ms 64 bytes
from 200.200.1.2: icmp_seq=4 ttl=254 time=44.6 ms 64 bytes
from 200.200.1.2: icmp_seq=5 ttl=254 time=44.7 ms 64 bytes
from 200.200.1.2: icmp_seq=6 ttl=254 time=44.6 ms 64 bytes
from 200.200.1.2: icmp_seq=7 ttl=254 time=44.6 ms 64 bytes
from 200.200.1.2: icmp_seq=8 ttl=254 time=44.6 ms 64 bytes
from 200.200.1.2: icmp_seq=9 ttl=254 time=44.6 ms
```

--- 200.200.1.2 ping statistics ---

9 packets transmitted, 9 received, 0% packet loss, time 8004ms

rtt min/avg/max/mdev = 44.583/44.649/44.772/0.269 ms

Comunicação entre o host 192.168.1.2 e a interface ethernet do roteador remoto:

```
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data. 64 bytes
from 192.168.2.1: icmp_seq=1 ttl=254 time=44.8 ms 64 bytes
from 192.168.2.1: icmp_seq=2 ttl=254 time=44.5 ms 64 bytes
from 192.168.2.1: icmp_seq=3 ttl=254 time=44.5 ms 64 bytes
from 192.168.2.1: icmp_seq=4 ttl=254 time=44.5 ms 64 bytes
from 192.168.2.1: icmp_seq=5 ttl=254 time=44.9 ms 64 bytes
from 192.168.2.1: icmp_seq=6 ttl=254 time=44.7 ms
```

--- 192.168.2.1 ping statistics ---

```
6 packets transmitted, 6 received, 0% packet loss, time 5003ms
rtt min/avg/max/mdev = 44.516/44.694/44.929/0.313 ms
```

Comunicação entre o host 192.168.1.2 e o host 192.168.2.2:

```
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data. 64 bytes
from 192.168.2.2: icmp_seq=1 ttl=62 time=45.3 ms 64 bytes
from 192.168.2.2: icmp_seq=2 ttl=62 time=44.8 ms 64 bytes
from 192.168.2.2: icmp_seq=3 ttl=62 time=44.4 ms 64 bytes
from 192.168.2.2: icmp_seq=4 ttl=62 time=44.7 ms
```

--- 192.168.2.2 ping statistics ---

```
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 44.491/44.868/45.346/0.345 ms
```

HOST 192.168.2.2

Comunicação entre o host 192.168.2.2 e a interface ethernet do roteador:

```
PING 192.168.2.1 (192.168.2.1) 56(84) bytes of data. 64 bytes
```

```
from 192.168.2.1: icmp_seq=1 ttl=255 time=1.46 ms 64 bytes
from 192.168.2.1: icmp_seq=2 ttl=255 time=1.06 ms 64 bytes
from 192.168.2.1: icmp_seq=3 ttl=255 time=1.07 ms 64 bytes
from 192.168.2.1: icmp_seq=4 ttl=255 time=1.06 ms 64 bytes
from 192.168.2.1: icmp_seq=5 ttl=255 time=1.02 ms 64 bytes
from 192.168.2.1: icmp_seq=6 ttl=255 time=1.03 ms 64 bytes
from 192.168.2.1: icmp_seq=7 ttl=255 time=1.07 ms
```

```
--- 192.168.2.1 ping statistics ---
```

```
7 packets transmitted, 7 received, 0% packet loss, time 6002ms
rtt min/avg/max/mdev = 1.026/1.115/1.460/0.142 ms
```

Comunicação entre o host 192.168.2.2 e a interface serial 0 do roteador:

```
PING 200.200.1.2 (200.200.1.2) 56(84) bytes of data. 64 bytes
from 200.200.1.2: icmp_seq=1 ttl=255 time=1.52 ms 64 bytes
from 200.200.1.2: icmp_seq=2 ttl=255 time=1.79 ms 64 bytes
from 200.200.1.2: icmp_seq=3 ttl=255 time=2.21 ms 64 bytes
from 200.200.1.2: icmp_seq=4 ttl=255 time=1.28 ms 64 bytes
from 200.200.1.2: icmp_seq=5 ttl=255 time=1.72 ms 64 bytes
from 200.200.1.2: icmp_seq=6 ttl=255 time=1.67 ms 64 bytes
from 200.200.1.2: icmp_seq=7 ttl=255 time=1.15 ms 64 bytes
from 200.200.1.2: icmp_seq=8 ttl=255 time=1.09 ms
```

```
--- 200.200.1.2 ping statistics ---
```

```
8 packets transmitted, 8 received, 0% packet loss, time 7003ms
rtt min/avg/max/mdev = 1.099/1.558/2.212/0.351 ms
```

Comunicação entre o host 192.168.2.2 e a interface serial 0 do roteador remoto:

```
PING 200.200.1.1 (200.200.1.1) 56(84) bytes of data. 64 bytes
from 200.200.1.1: icmp_seq=1 ttl=254 time=44.9 ms 64 bytes
from 200.200.1.1: icmp_seq=2 ttl=254 time=44.8 ms 64 bytes
```

```
from 200.200.1.1: icmp_seq=3 ttl=254 time=44.7 ms 64 bytes
from 200.200.1.1: icmp_seq=4 ttl=254 time=44.9 ms 64 bytes
from 200.200.1.1: icmp_seq=5 ttl=254 time=44.8 ms 64 bytes
from 200.200.1.1: icmp_seq=6 ttl=254 time=44.7 ms 64 bytes
from 200.200.1.1: icmp_seq=7 ttl=254 time=44.8 ms 64 bytes
from 200.200.1.1: icmp_seq=8 ttl=254 time=44.8 ms 64 bytes
from 200.200.1.1: icmp_seq=9 ttl=254 time=44.7 ms
```

```
--- 200.200.1.1 ping statistics ---
```

```
9 packets transmitted, 9 received, 0% packet loss, time 7999ms
rtt min/avg/max/mdev = 44.754/44.839/44.946/0.230 ms
```

Comunicação entre o host 192.168.2.2 e a interface ethernet do roteador remoto:

```
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data. 64 bytes
from 192.168.1.1: icmp_seq=1 ttl=254 time=45.0 ms 64 bytes
from 192.168.1.1: icmp_seq=2 ttl=254 time=45.5 ms 64 bytes
from 192.168.1.1: icmp_seq=3 ttl=254 time=45.0 ms 64 bytes
from 192.168.1.1: icmp_seq=4 ttl=254 time=44.8 ms 64 bytes
from 192.168.1.1: icmp_seq=5 ttl=254 time=44.7 ms 64 bytes
from 192.168.1.1: icmp_seq=6 ttl=254 time=44.9 ms 64 bytes
from 192.168.1.1: icmp_seq=7 ttl=254 time=44.9 ms 64 bytes
from 192.168.1.1: icmp_seq=8 ttl=254 time=44.9 ms 64 bytes
from 192.168.1.1: icmp_seq=9 ttl=254 time=44.9 ms 64 bytes
from 192.168.1.1: icmp_seq=10 ttl=254 time=44.7 ms
```

```
--- 192.168.1.1 ping statistics ---
```

```
10 packets transmitted, 10 received, 0% packet loss, time 9006ms
rtt min/avg/max/mdev = 44.739/44.972/45.522/0.264 ms
```

Comunicação entre o host 192.168.2.2 e o host 192.168.1.2:

```
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data. 64 bytes
```

```

from 192.168.1.2: icmp_seq=1 ttl=62 time=44.8 ms 64 bytes
from 192.168.1.2: icmp_seq=2 ttl=62 time=44.5 ms 64 bytes
from 192.168.1.2: icmp_seq=3 ttl=62 time=44.6 ms 64 bytes
from 192.168.1.2: icmp_seq=4 ttl=62 time=44.7 ms 64 bytes
from 192.168.1.2: icmp_seq=5 ttl=62 time=44.6 ms 64 bytes
from 192.168.1.2: icmp_seq=6 ttl=62 time=44.7 ms 64 bytes
from 192.168.1.2: icmp_seq=7 ttl=62 time=44.7 ms

```

```
--- 192.168.1.2 ping statistics ---
```

```

7 packets transmitted, 7 received, 0% packet loss, time 6006ms
rtt min/avg/max/mdev = 44.592/44.720/44.882/0.214 ms

```

Comunicação entre o host 192.168.2.2 e o host 192.168.1.3:

```

PING 192.168.1.3 (192.168.1.3) 56(84) bytes of data. 64 bytes
from 192.168.1.3: icmp_seq=1 ttl=62 time=88.8 ms 64 bytes
from 192.168.1.3: icmp_seq=2 ttl=62 time=52.7 ms 64 bytes
from 192.168.1.3: icmp_seq=3 ttl=62 time=48.0 ms 64 bytes
from 192.168.1.3: icmp_seq=4 ttl=62 time=44.8 ms 64 bytes
from 192.168.1.3: icmp_seq=5 ttl=62 time=50.6 ms 64 bytes
from 192.168.1.3: icmp_seq=6 ttl=62 time=46.5 ms

```

```
--- 192.168.1.3 ping statistics ---
```

```

7 packets transmitted, 6 received, 14% packet loss, time 6001ms
rtt min/avg/max/mdev = 44.852/55.286/88.834/15.225 ms

```

7.5.2 Traçando Rotas com o traceroute

Do host 192.168.2.2 até o host 192.168.1.2 da outra subrede, passando pelo seu gateway e também pelo roteador da rede remota.

```

1  192.168.2.1 (192.168.2.1)  1.166 ms  1.051 ms  1.021 ms
2  200.200.1.1 (200.200.1.1) 42.985 ms 44.596 ms 36.375 ms
3  192.168.1.2 (192.168.1.2) 37.577 ms 37.488 ms 37.510 ms

```

7.5.3 Capturando Pacotes com tcpdump

Comunicação entre o Host 192.168.2.2 e o interface ethernet do roteador da rede remota 192.168.1.1:

```
16:26:18.959625 IP 192.168.1.1 > 192.168.2.2: ICMP echo reply, id
47641, seq 101, length 64
 0x0000:  4500 0054 0000 4000 fe01 f854 c0a8 0101  E..T..@....T....
 0x0010:  c0a8 0202 0000 1f63 ba19 0065 4ae9 563c  .....c...eJ.V<
 0x0020:  8cf5 0d00 0809 0a0b 0c0d 0e0f 1011 1213  .....
 0x0030:  1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
 0x0040:  2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
 0x0050:  3435                                     45
```

```
16:26:19.915161 IP 192.168.2.2 > 192.168.1.1: ICMP echo request, id
47641, seq 102, length 64
 0x0000:  4500 0054 0000 4000 4001 b655 c0a8 0202  E..T..@..U....
 0x0010:  c0a8 0101 0800 e360 ba19 0066 4be9 563c  .....‘...fK.V<
 0x0020:  bff6 0d00 0809 0a0b 0c0d 0e0f 1011 1213  .....
 0x0030:  1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
 0x0040:  2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
 0x0050:  3435                                     45
```

```
16:26:19.959880 IP 192.168.1.1 > 192.168.2.2: ICMP echo reply, id
47641, seq 102, length 64
 0x0000:  4500 0054 0000 4000 fe01 f854 c0a8 0101  E..T..@....T....
 0x0010:  c0a8 0202 0000 eb60 ba19 0066 4be9 563c  .....‘...fK.V<
 0x0020:  bff6 0d00 0809 0a0b 0c0d 0e0f 1011 1213  .....
 0x0030:  1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
 0x0040:  2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
 0x0050:  3435                                     45
```

Comunicação entre os Hosts 192.168.2.2 e a interface serial 0 do roteador local:

```
16:27:04.341861 IP 192.168.2.2 > 200.200.1.2: ICMP echo request, id
```

49689, seq 9, length 64

```

0x0000: 4500 0054 0000 4000 4001 ae34 c0a8 0202 E..T..@.@..4....
0x0010: c8c8 0102 0800 2f7d c219 0009 78e9 563c ...../}....x.V<
0x0020: 4737 0500 0809 0a0b 0c0d 0e0f 1011 1213 G7.....
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
0x0050: 3435                                     45

```

16:27:04.342959 IP 200.200.1.2 > 192.168.2.2: ICMP echo reply, id

49689, seq 9, length 64

```

0x0000: 4500 0054 0000 4000 ff01 ef33 c8c8 0102 E..T..@....3....
0x0010: c0a8 0202 0000 377d c219 0009 78e9 563c .....7}....x.V<
0x0020: 4737 0500 0809 0a0b 0c0d 0e0f 1011 1213 G7.....
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
0x0050: 3435                                     45

```

Comunicação entre o Host 192.168.2.2 e a interface serial 0 do roteador da rede remota:

16:27:39.303385 IP 192.168.2.2 > 200.200.1.1: ICMP echo request, id

50713, seq 1, length 64

```

0x0000: 4500 0054 0000 4000 4001 ae35 c0a8 0202 E..T..@.@..5....
0x0010: c8c8 0101 0800 631b c619 0001 9be9 563c .....c.....V<
0x0020: eda0 0400 0809 0a0b 0c0d 0e0f 1011 1213 .....
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
0x0050: 3435                                     45

```

16:27:39.381214 IP 200.200.1.1 > 192.168.2.2: ICMP echo reply, id

50713, seq 1, length 64

```

0x0000: 4500 0054 0000 4000 fe01 f034 c8c8 0101 E..T..@....4....
0x0010: c0a8 0202 0000 6b1b c619 0001 9be9 563c .....k.....V<
0x0020: eda0 0400 0809 0a0b 0c0d 0e0f 1011 1213 .....
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
0x0050: 3435                                     45

```


Comunicação entre o Host 192.168.2.2 e o Host 192.168.1.2 da subrede remota:

```
16:28:19.895718 IP 192.168.2.2 > 192.168.1.2: ICMP echo request, id
51225, seq 5, length 64
 0x0000:  4500 0054 0000 4000 4001 b654 c0a8 0202  E..T..@.@..T....
 0x0010:  c0a8 0102 0800 560d c819 0005 c3e9 563c  .....V.....V<
 0x0020:  c7aa 0d00 0809 0a0b 0c0d 0e0f 1011 1213  .....
 0x0030:  1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
 0x0040:  2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
 0x0050:  3435                                     45

16:28:19.940185 IP 192.168.1.2 > 192.168.2.2: ICMP echo reply, id
51225, seq 5, length 64
 0x0000:  4500 0054 9b26 0000 3e01 5d2e c0a8 0102  E..T.&...>.].....
 0x0010:  c0a8 0202 0000 5e0d c819 0005 c3e9 563c  .....^.....V<
 0x0020:  c7aa 0d00 0809 0a0b 0c0d 0e0f 1011 1213  .....
 0x0030:  1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  .....!"#
 0x0040:  2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
 0x0050:  3435                                     45
```

7.6 Instalação e Configuração da Rede de Roteadores Usando Máquinas Virtuais

Objetivo:

- Criar uma malha com três roteadores interligando clientes de três redes distintas como mostrado na figura 27.

Atividades:

- Instalar e configurar máquinas virtuais como roteadores;
- Instalar e configurar máquinas virtuais como clientes;
- Criar e configurar redes entre os roteadores e entre clientes e roteadores.

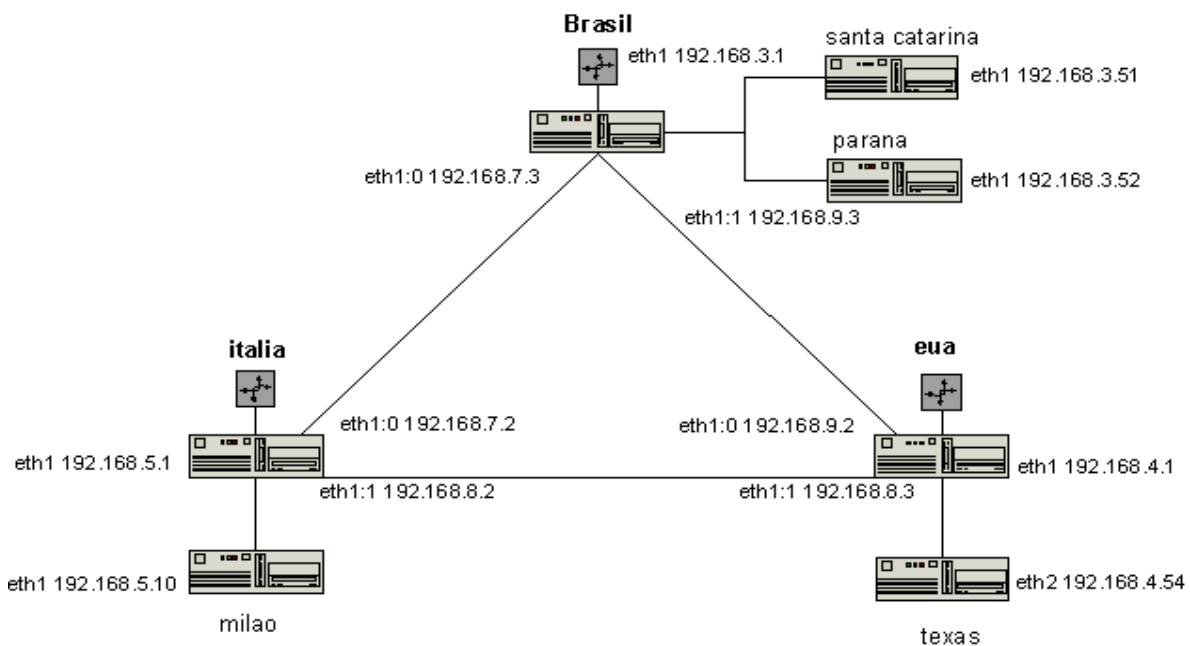


Figura 27: Diagrama de Redes de Roteadores.

Esta rede é composta por três roteadores: **brasil**, **eua** e **itália**. Cada roteador possui três interfaces de rede, sendo duas que os interligam e uma que liga cada roteador a sua rede local. Fisicamente há apenas uma interface de rede em cada roteador (a que o liga a rede local) e duas interfaces virtuais, cuja configuração será detalhada mais a frente. Não há a necessidade de switches na rede local 192.168.3.0/24 porque as interfaces estão configuradas como *bridge*. Ao final da configuração da rede de roteadores os clientes das redes locais poderão “enxergarem-se” e trocarem pacotes entre si.

Passo 01 - Instalação de 7 máquinas virtuais

A instalação das máquinas virtuais será padrão, com memória RAM de 32 MB, interface de rede configurada como “bridge” e disco rígido com 5 GB. O sistema operacional será o Linux Mandriva 2008. Sendo que na prática teremos um arquivo contendo os dados da máquina virtual, faremos apenas a instalação de uma máquina virtual e depois seis cópias desta.

Configuração das máquinas virtuais com o VMware

A configuração deve ser realizada de acordo com os passos descritos no anexo 7.2, respeitando os parâmetros citados acima.

A configuração do SO rodando na VMware é igual a que se faz na máquina sem a mesma. Instalar o servidor `ssh` e as ferramentas `tcpdump` e `traceroute`.

Passo 02 - Máquinas Roteadoras

Observação: todas as configurações listadas a seguir serão feitas em modo texto, sendo usado, quando necessário, o editor `vi`.

Configuração das interfaces de rede:

1. Digitar `ifconfig` para listar as interfaces existentes;

Exemplo de `ifconfig`:

```
eth1      Link encap:Ethernet  Endereço de HW 00:0C:29:70:F9:DD
          inet end.: 192.168.10.40  Bcast:192.168.10.255  Masc:255.255.255.0
          endereço inet6: fe80::20c:29ff:fe70:f9dd/64 Escopo:Link
          UP BROADCASTRUNNING MULTICAST  MTU:1500  Métrica:1
          RX packets:2809 errors:0 dropped:0 overruns:0 frame:0
          TX packets:170 errors:0 dropped:0 overruns:0 carrier:0
          colisões:0 txqueuelen:1000
          RX bytes:262266 (256.1 KiB)  TX bytes:33233 (32.4 KiB)
          IRQ:17 Endereço de E/S:0x1400

lo        Link encap:Loopback Local
          inet end.: 127.0.0.1  Masc:255.0.0.0
          endereço inet6: ::1/128 Escopo:Máquina
          UP LOOPBACKRUNNING  MTU:16436  Métrica:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          colisões:0 txqueuelen:0
          RX bytes:144 (144.0 b)  TX bytes:144 (144.0 b)
```

2. Configurar o arquivo **network**, digitando na linha de comando:

`vi /etc/sysconfig/network` onde serão definidos os nomes das máquinas, conforme dados da tabela 1. Este arquivo tem os seguintes parâmetros:

- NETWORKING = trabalha ou não em rede;
- GATEWAY = roteador padrão;
- GATEWAYDEV = interface de acesso ao roteador padrão;
- HOSTNAME = nome da máquina.

	Router1	Router2	Router3
NETWORKING	yes	yes	yes
HOSTNAME	brasil	eua	italia

Tabela 1: Configuração do arquivo `network`.

Obs.: Para este exemplo não é necessário editar o parâmetro `GATEWAYDEV`; o parâmetro `GATEWAY` será editado a seguir.

3. Configurar o arquivo `ifcfg-ethX`, onde X é o número da interface, conforme dados da tabela 2, digitando na linha de comando:

```
vi /etc/sysconfig/network-scripts/ifcfg-ethX.
```

Este arquivo tem os seguintes parâmetros:

- DEVICE = nome do dispositivo;
- BOOTPROTO = configuração estática ou dinâmica;
- IPADDR = endereço IP;
- NETMASK = máscara de rede;
- BROADCAST = endereço de broadcast;
- GATEWAY = roteador padrão;
- ONBOOT = interface inicializa no boot da máquina;
- MSDNS1 = endereço do servidor DNS primário;
- MSDNS2 = endereço do servidor DNS secundário.

	brasil	eua	italia
DEVICE	eth1	eth1	eth1
BOOTPROTO	static	static	static
IPADDR	192.168.3.1	192.168.4.1	192.168.5.1
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0
BROADCAST	192.168.3.255	192.168.4.255	192.168.5.255
GATEWAY	192.168.3.1	192.168.4.1	192.168.5.1
ONBOOT	yes	yes	yes

Tabela 2: Configuração do arquivo `ifcfg-ethX`.

Obs.: Para este exemplo não é necessário editar os parâmetros `MSDNS1` e `MSDNS2`.

4. Criar o arquivo `ifcfg-ethX:Y` para criar as interfaces virtuais (onde `X` é o número da interface real e `Y` é outro número, a partir de zero, para identificar as interfaces virtuais), conforme dados da tabela 3. Digitar na linha de comando:

`vi/etc/sysconfig/network-scripts/ifcfg-ethX:Y` e no editor `vi` digitar:

- `IPADDR` = endereço IP;
- `NETMASK` = máscara de rede;

	brasil	eua	italia
IPADDR	192.168.7.3 eth1:0	192.168.9.2 eth1:0	192.168.7.2 eth1:0
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0
IPADDR	192.168.9.3 eth1:1	192.168.8.3 eth1:1	192.168.8.2 eth1:1
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0

Tabela 3: Criação do arquivo `ifcfg-ethX:Y`.

Obs.: Após estas configurações reiniciar o serviço de rede com o comando: `service network restart`

Em nosso exemplo, todas as máquinas virtuais vão assumir a interface de rede física como sendo sua interface de rede. Como para os roteadores se faz necessário mais de uma interface de rede criamos as interfaces virtuais. Desta forma cada roteador terá três interfaces de rede, uma real e duas virtuais.

Configurando os roteadores:

1. Digitar o comando:

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

Desta forma estamos mudando o bit `ipforward` para 1 e tornando as máquinas roteadores.

2. Adicionando rotas:

Com o comando `route -n` é impresso na tela a tabela de roteamento atual.

roteador brasil

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usa	Iface
192.168.7.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.3.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.9.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
0.0.0.0	192.168.3.1	0.0.0.0	UG	10	0	0	eth1

roteador eua

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usa	Iface
192.168.4.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.9.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.8.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
0.0.0.0	192.168.4.1	0.0.0.0	UG	10	0	0	eth1

roteador italia

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usa	Iface
---------	----------	-------------	--------	---------	-----	-----	-------

```

192.168.7.0 0.0.0.0 255.255.255.0 U 10 0 0 eth1
192.168.5.0 0.0.0.0 255.255.255.0 U 10 0 0 eth1
192.168.8.0 0.0.0.0 255.255.255.0 U 10 0 0 eth1
0.0.0.0 192.168.5.1 0.0.0.0 UG 10 0 0 eth1

```

Após todas as configurações de rede descritas anteriormente essas serão as tabelas de roteamento padrão para cada roteador. A rede 192.168.7.0/24 liga o roteador **brasil** e **italia**, por isso aparece na tabela dos dois roteadores; a rede 192.168.9.0/24 liga os roteadores **brasil** e **eua**, aparecendo na tabela de roteamento de cada um e finalmente a rede 192.168.8.0/24 que liga os roteadores **italia** e **eua**. As redes 192.168.5.0/24, 192.168.4.0/24 e 192.168.3.0/24 correspondem as redes locais dos roteadores **italia**, **eua** e **brasil** respectivamente.

Para adicionar rotas digitamos:

```
route add -net 192.168.Y.0/24 gw 192.168.Y.Z
```

onde o primeiro endereço é o da rede que queremos atingir e o segundo endereço é o do roteador que selecionará a melhor rota para alcançar este objetivo. Para adicionar o roteador padrão que será usado quando não houver um roteador definido para determinada rede, digitamos:

```
route add default gw 192.168.Y.Z ethX
```

No roteador **brasil** vamos adicionar rotas para as redes 192.168.5.0/24, 192.168.4.0/24 e 192.168.8.0/24:

brasil

```

route add -net 192.168.5.0/24 gw 192.168.7.2
route add -net 192.168.4.0/24 gw 192.168.9.2
route add -net 192.168.8.0/24 gw 192.168.7.2
route add -net 192.168.8.0/24 gw 192.168.9.2
route add default gw 192.168.3.1 eth1

```

Verificar a tabela de roteamento : `route -n`

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usa	Iface
192.168.7.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.5.0	192.168.7.2	0.0.0.0	UG	0	0	0	eth1

192.168.4.0	192.168.9.2	255.255.255.0	UG	0	0	0	eth1
192.168.3.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.9.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.8.0	192.168.7.2	255.255.255.0	UG	0	0	0	eth1
192.168.8.0	192.168.9.2	255.255.255.0	UG	0	0	0	eth1
0.0.0.0	192.168.3.1	0.0.0.0	UG	10	0	0	eth1

No roteador **eua** vamos adicionar rotas para as redes 192.168.5.0/24, 192.168.3.0/24 e 192.168.7.0/24:

eua

```
route add -net 192.168.5.0/24 gw 192.168.8.2
route add -net 192.168.3.0/24 gw 192.168.9.3
route add -net 192.168.7.0/24 gw 192.168.8.2
route add -net 192.168.7.0/24 gw 192.168.9.3
route add default gw 192.168.4.1 eth1
```

Verificar a tabela de roteamento : `route -n`

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usa	Iface
192.168.7.0	192.168.8.2	255.255.255.0	U	10	0	0	eth1
192.168.7.0	192.168.9.3	255.255.255.0	UG	0	0	0	eth1
192.168.5.0	192.168.8.2	255.255.255.0	UG	0	0	0	eth1
192.168.4.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.3.0	192.168.9.3	255.255.255.0	UG	0	0	0	eth1
192.168.9.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.8.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
0.0.0.0	192.168.4.1	0.0.0.0	UG	10	0	0	eth1

No roteador **italia** vamos adicionar rotas para as redes 192.168.9.0/24, 192.168.4.0/24 e 192.168.3.0/24:

italia


```

route add -net 192.168.4.0/24 gw 192.168.8.2
route add -net 192.168.3.0/24 gw 192.168.9.3
route add -net 192.168.9.0/24 gw 192.168.8.2
route add -net 192.168.9.0/24 gw 192.168.9.3
route add default gw 192.168.5.1 eth1

```

Verificar a tabela de roteamento : `route -n`

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Uso	Iface
192.168.7.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.5.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.168.4.0	192.168.8.3	255.255.255.0	UG	0	0	0	eth1
192.168.3.0	192.168.7.3	255.255.255.0	UG	0	0	0	eth1
192.168.9.0	192.168.8.3	255.255.255.0	UG	0	0	0	eth1
192.168.9.0	192.168.7.3	255.255.255.0	UG	0	0	0	eth1
192.168.8.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
0.0.0.0	192.168.5.1	0.0.0.0	UG	10	0	0	eth1

Passo 03 - Máquinas Clientes

Configuração das interfaces de rede:

1. Digitar `ifconfig` para listar as interfaces existentes.
2. Configurar o arquivo `network`, digitando na linha de comando:
`vi /etc/sysconfig/network` onde serão definidos os nomes das máquinas, conforme dados da tabela 4. Este arquivo tem os seguintes parâmetros:

- NETWORKING = trabalha ou não em rede;
- GATEWAY = roteador padrão;
- GATEWAYDEV = interface de acesso ao roteador padrão;
- HOSTNAME = nome da máquina.

	Cliente1	Cliente2	Cliente3	Cliente4
NETWORKING	yes	yes	yes	yes
HOSTNAME	parana	santa catarina	texas	milao

Tabela 4: Configuração do arquivo `network`.

Obs.: Para este exemplo não é necessário editar o parâmetro `GATEWAYDEV`; o parâmetro `GATEWAY` será editado a seguir.

3. Configurar o arquivo `ifcfg-ethX`, onde X é o número da interface, conforme dados da tabela 5, digitando na linha de comando:

```
vi/etc/sysconfig/network-scripts/ifcfg-ethX.
```

Este arquivo tem os seguintes parâmetros:

- DEVICE = nome do dispositivo;
- BOOTPROTO = configuração estática ou dinâmica;
- IPADDR = endereço IP;
- NETMASK = máscara de rede;
- BROADCAST = endereço de broadcast;
- GATEWAY = roteador padrão;
- ONBOOT = interface inicializa no boot da máquina;
- MSDNS1 = endereço do servidor DNS primário;
- MSDNS2 = endereço do servidor DNS secundário.

Observações:

- Para este exemplo não é necessário editar os parâmetros `MSDNS1` e `MSDNS2`;
- Após estas configurações reiniciar o serviço de rede: `service network restart`.

4. Verificando a tabela de roteamento das máquinas clientes: `route -n`

parana

	parana	santa catarina	texas	milao
DEVICE	eth1	eth1	eth2	eth0
BOOTPROTO	static	static	static	static
IPADDR	192.168.3.52	192.168.3.51	192.168.4.54	192.168.5.10
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
BROADCAST	192.168.3.255	192.168.3.255	192.168.4.255	192.168.5.255
GATEWAY	192.168.3.1	192.168.3.1	192.168.4.1	192.168.5.1
ONBOOT	yes	yes	yes	yes

Tabela 5: Configuração do arquivo ifcfg-ethX.

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usos	Iface
192.168.3.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
0.0.0.0	192.168.3.1	0.0.0.0	UG	10	0	0	eth1

santa catarina

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usos	Iface
192.168.3.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
0.0.0.0	192.168.3.1	0.0.0.0	UG	10	0	0	eth1

texas

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usos	Iface
192.168.4.0	0.0.0.0	255.255.255.0	U	10	0	0	eth2
0.0.0.0	192.168.4.1	0.0.0.0	UG	10	0	0	eth2

milao

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usos	Iface
192.168.5.0	0.0.0.0	255.255.255.0	U	10	0	0	eth0
0.0.0.0	192.168.5.1	0.0.0.0	UG	10	0	0	eth0

A tabela de roteamento das máquinas clientes mostra apenas a rede a que a máquina está ligada e o roteador padrão desta rede. Para trocar pacotes com outro cliente das outras redes locais será utilizado apenas o roteador padrão, sendo então nas máquinas roteadoras feito o encaminhamento destes pacotes . Os testes realizados são apresentados abaixo.

7.6.1 Testes com PING

PING da VM (cliente) parana na rede 192.168.3.0/24 para a VM (cliente) milao na rede 192.168.5.0/24

```
PING 192.168.5.10 (192.168.5.10) 56(84) bytes of data. 64 bytes
from 192.168.5.10: icmp_seq=1 ttl=62 time=90.5 ms 64 bytes
from 192.168.5.10: icmp_seq=2 ttl=62 time=15.7 ms 64 bytes
from 192.168.5.10: icmp_seq=3 ttl=62 time=21.8 ms 64 bytes
from 192.168.5.10: icmp_seq=4 ttl=62 time=12.1 ms 64 bytes
from 192.168.5.10: icmp_seq=5 ttl=62 time=14.2 ms 64 bytes
from 192.168.5.10: icmp_seq=6 ttl=62 time=14.9 ms 64 bytes
from 192.168.5.10: icmp_seq=7 ttl=62 time=18.2 ms 64 bytes
from 192.168.5.10: icmp_seq=8 ttl=62 time=12.4 ms 64 bytes
from 192.168.5.10: icmp_seq=9 ttl=62 time=14.9 ms 64 bytes
from 192.168.5.10: icmp_seq=10 ttl=62 time=19.5 ms 64 bytes
from 192.168.5.10: icmp_seq=11 ttl=62 time=15.9 ms
```

```
--- 192.168.5.10 ping statistics ---
```

```
11 packets transmitted, 11 received, 0% packet loss, time 10115ms
```

```
rtt min/avg/max/mdev = 12.142/22.781/90.549/21.609 ms
```

PING da VM (cliente) parana na rede 192.168.3.0/24 para a VM (cliente) texas na rede 192.168.4.0/24

```
PING 192.168.4.54 (192.168.4.54) 56(84) bytes of data. 64 bytes
```

```
from 192.168.4.54: icmp_seq=1 ttl=62 time=58.7 ms 64 bytes
from 192.168.4.54: icmp_seq=2 ttl=62 time=16.1 ms 64 bytes
from 192.168.4.54: icmp_seq=3 ttl=63 time=14.2 ms 64 bytes
from 192.168.4.54: icmp_seq=4 ttl=63 time=10.1 ms 64 bytes
from 192.168.4.54: icmp_seq=5 ttl=63 time=10.3 ms 64 bytes
from 192.168.4.54: icmp_seq=6 ttl=63 time=12.9 ms 64 bytes
from 192.168.4.54: icmp_seq=7 ttl=63 time=15.0 ms
```

```
--- 192.168.4.54 ping statistics ---
```

```
7 packets transmitted, 7 received, 0% packet loss, time 6053ms
rtt min/avg/max/mdev = 10.107/19.642/58.765/16.110 ms
```

PING da VM (cliente) santa catarina na rede 192.168.3.0/24 para a VM (cliente) milao na rede 192.168.5.0/24

```
PING 192.168.5.10 (192.168.5.10) 56(84) bytes of data. 64 bytes
from 192.168.5.10: icmp_seq=1 ttl=62 time=5.27 ms 64 bytes
from 192.168.5.10: icmp_seq=2 ttl=62 time=14.5 ms 64 bytes
from 192.168.5.10: icmp_seq=3 ttl=62 time=13.5 ms 64 bytes
from 192.168.5.10: icmp_seq=4 ttl=62 time=19.7 ms 64 bytes
from 192.168.5.10: icmp_seq=5 ttl=62 time=21.5 ms 64 bytes
from 192.168.5.10: icmp_seq=6 ttl=62 time=17.1 ms 64 bytes
from 192.168.5.10: icmp_seq=7 ttl=62 time=35.9 ms
```

```
--- 192.168.5.10 ping statistics ---
```

```
7 packets transmitted, 7 received, 0% packet loss, time 6010ms
rtt min/avg/max/mdev = 5.279/18.250/35.901/8.702 ms
```

PING da VM (cliente) santa catarina na rede 192.168.3.0/24 para a VM (cliente) texas na rede 192.168.4.0/24

```
PING 192.168.4.54 (192.168.4.54) 56(84) bytes of data. 64 bytes
```

```
from 192.168.4.54: icmp_seq=1 ttl=62 time=84.8 ms 64 bytes
from 192.168.4.54: icmp_seq=2 ttl=62 time=7.26 ms 64 bytes
from 192.168.4.54: icmp_seq=3 ttl=63 time=7.97 ms 64 bytes
from 192.168.4.54: icmp_seq=4 ttl=63 time=9.22 ms 64 bytes
from 192.168.4.54: icmp_seq=5 ttl=63 time=9.16 ms 64 bytes
from 192.168.4.54: icmp_seq=6 ttl=63 time=15.0 ms 64 bytes
from 192.168.4.54: icmp_seq=7 ttl=63 time=13.8 ms
```

```
--- 192.168.4.54 ping statistics ---
```

```
7 packets transmitted, 7 received, 0% packet loss, time 6003ms
rtt min/avg/max/mdev = 7.264/21.061/84.854/26.187 ms
```

PING da VM (cliente) milao na rede 192.168.5.0/24 para a VM (cliente) texas na rede 192.168.4.0/24

```
PING 192.168.4.54 (192.168.4.54) 56(84) bytes of data. From
192.168.5.1: icmp_seq=1 Redirect Host(New nexthop: 192.168.8.3) 64
bytes from 192.168.4.54: icmp_seq=1 ttl=63 time=36.0 ms From
192.168.5.1: icmp_seq=2 Redirect Host(New nexthop: 192.168.8.3) 64
bytes from 192.168.4.54: icmp_seq=2 ttl=63 time=14.3 ms From
192.168.5.1: icmp_seq=3 Redirect Host(New nexthop: 192.168.8.3) 64
bytes from 192.168.4.54: icmp_seq=3 ttl=63 time=28.9 ms From
192.168.5.1: icmp_seq=4 Redirect Host(New nexthop: 192.168.8.3) 64
bytes from 192.168.4.54: icmp_seq=4 ttl=63 time=13.0 ms From
192.168.5.1: icmp_seq=5 Redirect Host(New nexthop: 192.168.8.3) 64
bytes from 192.168.4.54: icmp_seq=5 ttl=63 time=23.3 ms From
192.168.5.1: icmp_seq=6 Redirect Host(New nexthop: 192.168.8.3) 64
bytes from 192.168.4.54: icmp_seq=6 ttl=63 time=10.7 ms From
192.168.5.1: icmp_seq=7 Redirect Host(New nexthop: 192.168.8.3) 64
bytes from 192.168.4.54: icmp_seq=7 ttl=63 time=35.1 ms From
192.168.5.1: icmp_seq=8 Redirect Host(New nexthop: 192.168.8.3) 64
bytes from 192.168.4.54: icmp_seq=8 ttl=63 time=9.60 ms 64 bytes
from 192.168.4.54: icmp_seq=9 ttl=63 time=10.0 ms From 192.168.5.1:
icmp_seq=10 Redirect Host(New nexthop: 192.168.8.3) 64 bytes from
```

```
192.168.4.54: icmp_seq=10 ttl=63 time=55.5 ms 64 bytes
from 192.168.4.54: icmp_seq=11 ttl=63 time=4.01 ms 64 bytes
from 192.168.4.54: icmp_seq=12 ttl=63 time=9.71 ms 64 bytes
from 192.168.4.54: icmp_seq=13 ttl=63 time=9.78 ms 64 bytes
from 192.168.4.54: icmp_seq=14 ttl=63 time=11.9 ms 64 bytes
from 192.168.4.54: icmp_seq=15 ttl=63 time=13.8 ms 64 bytes
from 192.168.4.54: icmp_seq=16 ttl=63 time=11.4 ms 64 bytes
from 192.168.4.54: icmp_seq=17 ttl=63 time=14.0 ms
```

```
--- 192.168.4.54 ping statistics ---
```

```
17 packets transmitted, 17 received, 0% packet loss, time 16027ms
rtt min/avg/max/mdev = 4.014/18.336/55.571/12.942 ms
```

PING da VM (cliente) milao na rede 192.168.5.0/24 para a VM (cliente) santa catarina na rede 192.168.3.0/24

```
PING 192.168.3.51 (192.168.3.51) 56(84) bytes of data. From
192.168.5.1: icmp_seq=1 Redirect Host(New nexthop: 192.168.7.3) 64
bytes from 192.168.3.51: icmp_seq=1 ttl=63 time=174 ms From
192.168.5.1: icmp_seq=2 Redirect Host(New nexthop: 192.168.7.3) 64
bytes from 192.168.3.51: icmp_seq=2 ttl=63 time=9.34 ms From
192.168.5.1: icmp_seq=3 Redirect Host(New nexthop: 192.168.7.3) 64
bytes from 192.168.3.51: icmp_seq=3 ttl=63 time=114 ms From
192.168.5.1: icmp_seq=4 Redirect Host(New nexthop: 192.168.7.3) 64
bytes from 192.168.3.51: icmp_seq=4 ttl=63 time=9.18 ms From
192.168.5.1: icmp_seq=5 Redirect Host(New nexthop: 192.168.7.3) 64
bytes from 192.168.3.51: icmp_seq=5 ttl=63 time=28.0 ms From
192.168.5.1: icmp_seq=6 Redirect Host(New nexthop: 192.168.7.3) 64
bytes from 192.168.3.51: icmp_seq=6 ttl=63 time=40.7 ms From
192.168.5.1: icmp_seq=7 Redirect Host(New nexthop: 192.168.7.3) 64
bytes from 192.168.3.51: icmp_seq=7 ttl=63 time=136 ms From
192.168.5.1: icmp_seq=8 Redirect Host(New nexthop: 192.168.7.3) 64
bytes from 192.168.3.51: icmp_seq=8 ttl=63 time=8.79 ms 64 bytes
from 192.168.3.51: icmp_seq=9 ttl=63 time=17.3 ms From 192.168.5.1:
```

```
icmp_seq=10 Redirect Host(New nexthop: 192.168.7.3) 64 bytes
from 192.168.3.51: icmp_seq=10 ttl=63 time=225 ms 64 bytes
from 192.168.3.51: icmp_seq=11 ttl=63 time=13.5 ms 64 bytes
from 192.168.3.51: icmp_seq=12 ttl=63 time=6.06 ms 64 bytes
from 192.168.3.51: icmp_seq=13 ttl=63 time=15.2 ms 64 bytes
from 192.168.3.51: icmp_seq=14 ttl=63 time=36.9 ms 64 bytes
from 192.168.3.51: icmp_seq=15 ttl=63 time=29.6 ms 64 bytes
from 192.168.3.51: icmp_seq=16 ttl=63 time=7.80 ms
```

```
--- 192.168.3.51 ping statistics ---
```

```
16 packets transmitted, 16 received, 0% packet loss, time 15029ms
rtt min/avg/max/mdev = 6.060/54.644/225.824/66.694 ms
```

PING da VM (cliente) texas na rede 192.168.4.0/24 para a VM (cliente) milao na rede 192.168.5.0/24

```
PING 192.168.5.10 (192.168.5.10) 56(84) bytes of data.
```

```
From 192.168.4.1: icmp_seq=1 Redirect Host(New nexthop: 192.168.8.2)
64 bytes from 192.168.5.10: icmp_seq=1 ttl=62 time=25.3 ms
```

```
From 192.168.4.1: icmp_seq=2 Redirect Host(New nexthop: 192.168.8.2)
64 bytes
```

```
from 192.168.5.10: icmp_seq=2 ttl=62 time=10.2 ms 64 bytes
```

```
from 192.168.5.10: icmp_seq=3 ttl=62 time=20.8 ms 64 bytes
```

```
from 192.168.5.10: icmp_seq=4 ttl=62 time=5.01 ms 64 bytes
```

```
from 192.168.5.10: icmp_seq=5 ttl=62 time=23.1 ms 64 bytes
```

```
from 192.168.5.10: icmp_seq=6 ttl=62 time=21.0 ms 64 bytes
```

```
from 192.168.5.10: icmp_seq=7 ttl=62 time=17.3 ms
```

```
--- 192.168.5.10 ping statistics ---
```

```
7 packets transmitted, 7 received, 0% packet loss, time 6010ms
rtt min/avg/max/mdev = 5.011/17.555/25.310/6.814 ms
```


PING da VM (cliente) texas na rede 192.168.4.0/24 para a VM (cliente) parana na rede 192.168.5.0/24

```
PING 192.168.3.52 (192.168.3.52) 56(84) bytes of data.  
From 192.168.4.1: icmp_seq=1 Redirect Host(New nexthop: 192.168.9.3)  
64 bytes  
from 192.168.3.52: icmp_seq=1 ttl=62 time=100 ms  
From 192.168.4.1: icmp_seq=2 Redirect Host(New nexthop: 192.168.9.3)  
64 bytes from 192.168.3.52: icmp_seq=2 ttl=62 time=5.58 ms 64 bytes  
from 192.168.3.52: icmp_seq=3 ttl=62 time=37.6 ms 64 bytes  
from 192.168.3.52: icmp_seq=4 ttl=62 time=30.7 ms 64 bytes  
from 192.168.3.52: icmp_seq=5 ttl=62 time=12.9 ms 64 bytes  
from 192.168.3.52: icmp_seq=6 ttl=62 time=33.2 ms  
  
--- 192.168.3.52 ping statistics ---  
6 packets transmitted, 6 received, 0% packet loss, time 5026ms  
rtt min/avg/max/mdev = 5.589/36.747/100.316/30.628 ms
```

7.6.2 Traçando Rotas com o traceroute

Da VM (cliente) texas para a VM (cliente) milao

```
1 192.168.8.2 (192.168.8.2) 35.356 ms 8.084 ms 11.017 ms  
2 192.168.5.10 (192.168.5.10) 54.527 ms 23.948 ms 16.031 ms
```

Da VM (cliente) texas para a VM (cliente) parana

```
1 192.168.9.3 (192.168.9.3) 19.977 ms 12.578 ms 8.106 ms  
2 192.168.3.52 (192.168.3.52) 36.843 ms 42.848 ms 52.462 ms
```

Da VM (cliente) santa catarina para a VM (cliente) milao

1	192.168.3.1 (192.168.3.1)	71.601 ms	4.758 ms	3.241 ms
2	192.168.7.2 (192.168.7.2)	54.194 ms	6.107 ms	8.291 ms
3	192.168.5.10 (192.168.5.10)	30.989 ms	12.016 ms	14.927 ms

Da VM (cliente) parana para a VM (cliente) texas

1	192.168.3.1 (192.168.3.1)	28.349 ms	14.600 ms	6.947 ms
2	192.168.9.2 (192.168.9.2)	15.640 ms	4.725 ms	4.353 ms
3	192.168.4.54 (192.168.4.54)	4.139 ms	28.685 ms	12.696 ms

Da VM (cliente) parana para a VM (cliente) milao

1	192.168.3.1 (192.168.3.1)	22.074 ms	12.629 ms	6.455 ms
2	192.168.7.2 (192.168.7.2)	10.298 ms	69.737 ms	33.992 ms
3	192.168.5.10 (192.168.5.10)	58.711 ms	18.839 ms	36.846 ms

Da VM (cliente) milao para a VM (cliente) texas

1	192.168.5.1 (192.168.5.1)	5.633 ms	4.393 ms	3.339 ms
2	192.168.8.3 (192.168.8.3)	235.982 ms	295.469 ms	237.695 ms
3	192.168.4.54 (192.168.4.54)	174.193 ms	190.006 ms	173.831 ms

Da VM (cliente) milao para a VM (cliente) parana

1	192.168.5.1 (192.168.5.1)	8.186 ms	3.312 ms	1.747 ms
2	192.168.7.3 (192.168.7.3)	13.579 ms	68.134 ms	72.437 ms

3 192.168.3.52 (192.168.3.52) 213.571 ms 69.512 ms 48.949 ms

Da VM (roteador) italia para a VM (cliente) parana

1 192.168.7.3 (192.168.7.3) 69.333 ms 32.795 ms 26.564 ms
2 192.168.3.52 (192.168.3.52) 161.969 ms 26.274 ms 35.144 ms

Da VM (roteador) italia para a VM (cliente) texas

1 192.168.8.3 (192.168.8.3) 4.820 ms 10.486 ms 17.375 ms
2 192.168.4.54 (192.168.4.54) 108.253 ms 19.488 ms 36.266 ms

Da VM (roteador) brasil para a VM (cliente) milao

1 192.168.7.2 (192.168.7.2) 1.987 ms 10.930 ms 8.189 ms
2 192.168.5.10 (192.168.5.10) 26.854 ms 27.497 ms 10.508 ms

Da VM (roteador) brasil para a VM (cliente) texas

1 192.168.9.2 (192.168.9.2) 12.080 ms 2.127 ms 1.667 ms
2 192.168.4.54 (192.168.4.54) 6.133 ms 4.809 ms 6.007 ms

Da VM (roteador) eua para a VM (cliente) milao

1 192.168.8.2 (192.168.8.2) 5.832 ms 1.575 ms 1.252 ms
2 192.168.5.10 (192.168.5.10) 10.164 ms 9.235 ms 2.300 ms

7.6.3 Capturando Pacotes com tcpdump

Captura de pacote na interface eth1 da VM (roteador) brasil

```
IP 192.168.1.71.1088 > 234.8.7.1.1088: UDP, length 64 IP
192.168.5.10 > 192.168.3.1: ICMP echo request, id 1045, seq 23,
length 64 IP 192.168.5.10 > 192.168.3.1: ICMP echo request, id 1045,
seq 23, length 64 IP 192.168.3.1 > 192.168.5.10: ICMP echo reply, id
1045, seq 23, length 64 IP 192.168.3.1 > 192.168.5.10: ICMP echo
reply, id 1045, seq 23, length 64 IP 192.168.1.71.1088 >
234.8.7.1.1088: UDP, length 64 STP 802.1d, Config, Flags [none],
bridge-id 8000.00:0d:54:67:ad:80.8004, length 43
```

Captura de pacote na interface eth1 da VM (cliente) parana

```
IP 192.168.1.71.1088 > 234.8.7.1.1088: UDP, length 64 IP
192.168.4.54 > 192.168.3.52: ICMP echo request, id 32032, seq 52,
length 64 IP 192.168.4.54 > 192.168.3.52: ICMP echo request, id
32032, seq 52, length 64 IP 192.168.3.52 > 192.168.4.54: ICMP echo
reply, id 32032, seq 52, length 64 IP 192.168.3.52 > 192.168.4.54:
ICMP echo reply, id 32032, seq 52, length 64 IP 192.168.3.52 >
192.168.4.54: ICMP echo reply, id 32032, seq 52, length 64 IP
192.168.1.71.1088 > 234.8.7.1.1088: UDP, length 64
```

Captura de pacote na interface eth2 da VM (cliente) texas

```
IP 192.168.8.2 > 192.168.4.54: ICMP echo request, id 38944, seq 539,
length 64 IP 192.168.8.2 > 192.168.4.54: ICMP echo request, id
38944, seq 539, length 64 IP 192.168.4.54 > 192.168.8.2: ICMP echo
reply, id 38944, seq 539, length 64 IP 192.168.1.71.1088 >
234.8.7.1.1088: UDP, length 64 arp who-has 192.168.4.2 tell
192.168.4.54 IP 192.168.8.2 > 192.168.4.54: ICMP echo request, id
38944, seq 540, length 64 IP 192.168.8.2 > 192.168.4.54: ICMP echo
request, id 38944, seq 540, length 64
```

7.7 Configuração de uma Rede Virtual com Acesso a Internet no Laboratório de Redes I

Objetivo:

- Criar uma rede virtual com acesso à Internet no laboratório de redes I usando a infra-estrutura do mesmo. Não será alterada a infra-estrutura nem as configurações da rede do laboratório como mostrado na figura 28.

Infra-estrutura:

- O Laboratório de Redes I possui 17 máquinas reais conectadas a um *switch* de 24 portas, que por sua vez está conectado a rede do CEFET/SC - Unidade São José. Cada máquina real possui configurada uma máquina virtual com sistema operacional Linux Mandriva 2008 instalado. Como as interfaces de rede das máquinas virtuais estão configuradas como *bridged* (conectadas diretamente a rede física) teremos uma rede virtual de 17 máquinas conectadas ao switch de 24 portas.

Atividades:

- Configurar as interfaces de rede das máquinas virtuais já instaladas nas máquinas reais;
- Criar uma interface de rede virtual na máquina “mateus” (micro do professor) e torná-lo roteador da rede virtual provendo acesso a Internet

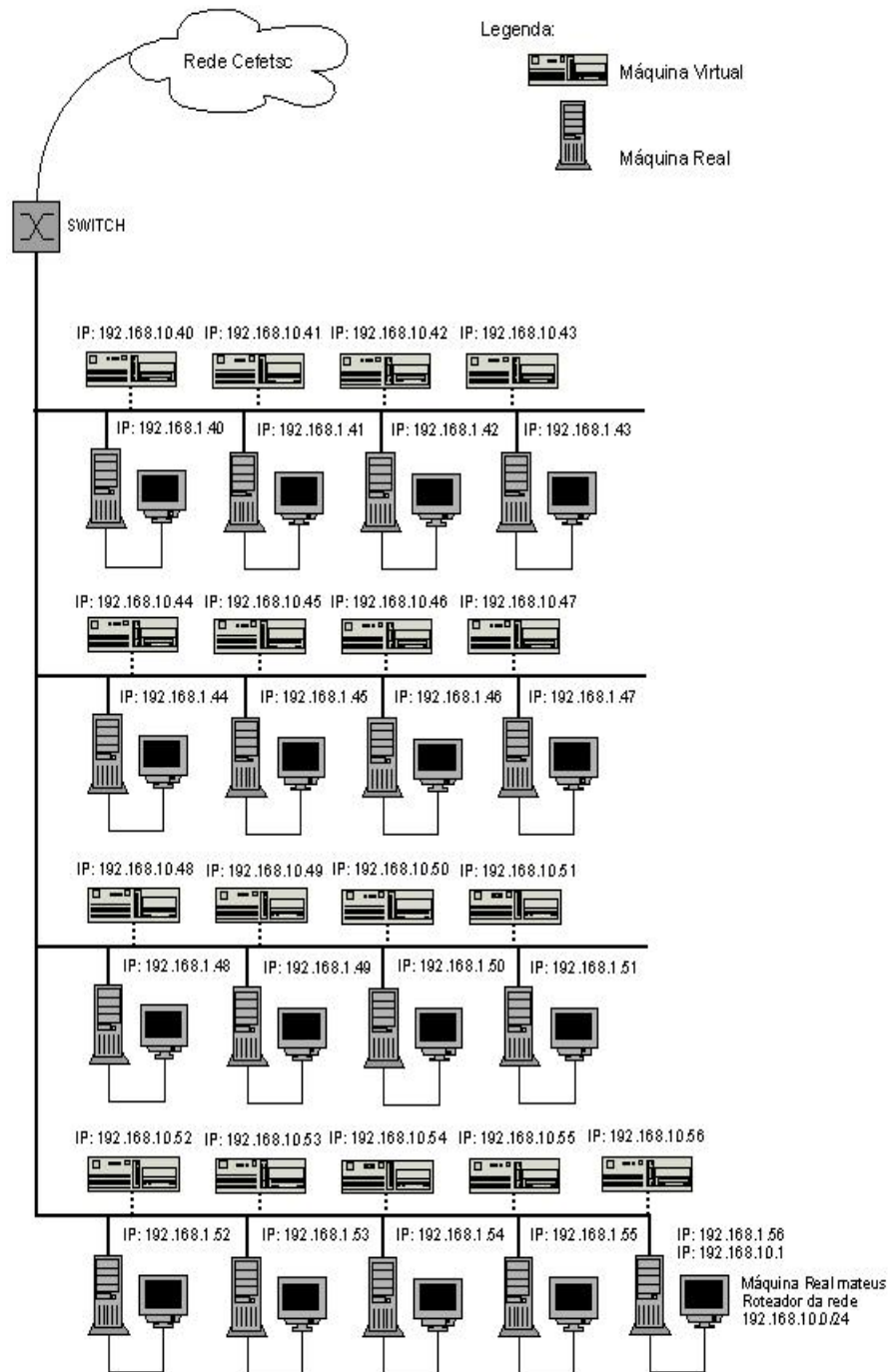


Figura 28: Diagrama de Rede Virtual com Acesso a Internet no Laboratório de Redes I.

Passo 01 - Configuração das interfaces de rede das máquinas virtuais

1. Digitar `ifconfig` para listar as interfaces existentes;

Exemplo de ifconfig:

```
eth1      Link encap:Ethernet  Endereço de HW 00:0C:29:70:F9:DD
          inet end.: 192.168.10.40  Bcast:192.168.10.255  Masc:255.255.255.0
          endereço inet6: fe80::20c:29ff:fe70:f9dd/64 Escopo:Link
          UP BROADCASTRUNNING MULTICAST  MTU:1500  Métrica:1
          RX packets:2809 errors:0 dropped:0 overruns:0 frame:0
          TX packets:170 errors:0 dropped:0 overruns:0 carrier:0
          colisões:0 txqueuelen:1000
          RX bytes:262266 (256.1 KiB)  TX bytes:33233 (32.4 KiB)
          IRQ:17  Endereço de E/S:0x1400

lo        Link encap:Loopback Local
          inet end.: 127.0.0.1  Masc:255.0.0.0
          endereço inet6: ::1/128 Escopo:Máquina
          UP LOOPBACKRUNNING  MTU:16436  Métrica:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          colisões:0 txqueuelen:0
          RX bytes:144 (144.0 b)  TX bytes:144 (144.0 b)
```

2. Configurar o arquivo `network`, digitando na linha de comando:

`vi /etc/sysconfig/network` onde serão definidos os nomes das máquinas, conforme dados da tabela 6. Este arquivo tem os seguintes parâmetros:

- `NETWORKING` = trabalha ou não em rede;
- `GATEWAY` = roteador padrão;
- `GATEWAYDEV` = interface de acesso ao roteador padrão;
- `HOSTNAME` = nome da máquina;

Para os roteadores será editado da seguinte forma :

	Maquina1	Maquina2	Maquina3	Maquina4	Maquina5	Maquina6
NETWORKING	yes	yes	yes	yes	yes	yes
HOSTNAME	italia	luxemburgo	noruega	suecia	irlanda	alemanha
	Maquina7	Maquina8	Maquina9	Maquina10	Maquina11	Maquina12
NETWORKING	yes	yes	yes	yes	yes	yes
HOSTNAME	portugal	suica	belgica	escocia	franca	austria
	Maquina13	Maquina14	Maquina15	Maquina16	Maquina17	
NETWORKING	yes	yes	yes	yes	yes	
HOSTNAME	grecia	espanha	dinamarca	monaco	inglaterra	

Tabela 6: Configuração do arquivo *network*

Obs.: Para este exemplo não é necessário editar o parâmetro `GATEWAYDEV` e o parâmetro `GATEWAY` será editado a seguir.

3. Configurar o arquivo `ifcfg-ethX`, onde X é o número da interface, conforme dados da tabela 7, digitando na linha de comando:

```
vi /etc/sysconfig/network-scripts/ifcfg-ethX
```

Este arquivo tem os seguintes parâmetros:

- `DEVICE` = nome do dispositivo;
- `BOOTPROTO` = configuração estática ou dinâmica;
- `IPADDR` = endereço IP;
- `NETMASK` = máscara de rede;
- `BROADCAST` = endereço de broadcast;
- `GATEWAY` = roteador padrão;
- `ONBOOT` = interface inicializa no boot da máquina;
- `MSDNS1` = endereço do servidor DNS primário;
- `MSDNS2` = endereço do servidor DNS secundário.

	italia	belgica	luxemburgo	inglaterra	franca
DEVICE	eth0	eth0	eth0	eth0	eth0
BOOTPROTO	static	static	static	static	static
IPADDR	192.168.10.40	192.168.10.41	192.168.10.42	192.168.10.43	192.168.10.44
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
BROADCAST	192.168.10.255	192.168.10.255	192.168.10.255	192.168.10.255	192.168.10.255
GATEWAY	192.168.10.1	192.168.10.1	192.168.10.1	192.168.10.1	192.168.10.1
ONBOOT	yes	yes	yes	yes	yes
	dinamarca	noruega	suecia	suica	portugal
DEVICE	eth0	eth0	eth0	eth0	eth0
BOOTPROTO	static	static	static	static	static
IPADDR	192.168.10.45	192.168.10.46	192.168.10.47	192.168.10.48	192.168.10.49
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
BROADCAST	192.168.10.255	192.168.10.255	192.168.10.255	192.168.10.255	192.168.10.255
GATEWAY	192.168.10.1	192.168.10.1	192.168.10.1	192.168.10.1	192.168.10.1
ONBOOT	yes	yes	yes	yes	yes
	espanha	grecia	alemanha	austria	irlanda
DEVICE	eth0	eth0	eth0	eth0	eth0
BOOTPROTO	static	static	static	static	static
IPADDR	192.168.10.50	192.168.10.51	192.168.10.52	192.168.10.53	192.168.10.54
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
BROADCAST	192.168.10.255	192.168.10.255	192.168.10.255	192.168.10.255	192.168.10.255
GATEWAY	192.168.10.1	192.168.10.1	192.168.10.1	192.168.10.1	192.168.10.1
ONBOOT	yes	yes	yes	yes	yes
	escocia	monaco			
DEVICE	eth0	eth0			
BOOTPROTO	static	static			
IPADDR	192.168.10.55	192.168.10.56			
NETMASK	255.255.255.0	255.255.255.0			
BROADCAST	192.168.10.255	192.168.10.255			
GATEWAY	192.168.10.1	192.168.10.1			
ONBOOT	yes	yes			

Tabela 7: Configuração do arquivo ifcfg-ethX.

Observações:

- Para este exemplo não é necessário editar os parâmetros `MSDNS1` e `MSDNS2`;
- Após estas configurações reiniciar o serviço de rede: `service network restart`;
- Todas estas configurações podem ser realizadas digitando `drakconnect` na linha de comando e preenchendo e/ou marcando os campos solicitados;
- Desabilitar o `firewall` do Mandriva digitando `drakfirewall` e marcando a opção que desabilita o `firewall` e depois marcar as interfaces de rede existentes.

A rede virtual terá o endereço 192.168.10.0/24.

Passo 02 - Configuração da máquina mateus

1. Criar uma interface de rede virtual na máquina “mateus”.

Criar o arquivo `ifcfg-ethX:Y` para criar as interfaces virtuais (onde X é o número da interface real e Y é outro número, a partir de zero, para identificar as interfaces virtuais). Digitar na linha de comando:

`vi/etc/sysconfig/network-scripts/ifcfg-ethX:Y` e no editor `vi` digitar:

- `IPADDR` = endereço IP;
- `NETMASK` = máscara de rede.

Na máquina real “mateus” a interface virtual terá a configuração:

- `IPADDR` = 192.168.10.1;
- `NETMASK` = 255.255.255.0.

Obs.: Após esta configuração reiniciar o serviço de rede com o comando: `service network restart`

2. Configurar a máquina real “mateus” como roteador para rede 192.168.10.0/24

Digitar o comando:

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

Desta forma estamos mudando o bit `ipforward` para 1 e tornando as máquinas roteadores.

Para adicionar rotas digitamos:

```
route add -net 192.168.Y.0/24 gw 192.168.Y.Z
```

onde o primeiro endereço é o da rede que queremos atingir e o segundo endereço é o do roteador que selecionará a melhor rota para alcançar este objetivo:

```
route add -net 192.168.10.0/24 gw 192.168.10.1
```

Configurar a máquina real “mateus” para fazer mascaramento da rede 192.168.10.0/24:

```
iptables -t nat -A POSTROUTING -s 192.168.10.0/24 -o eth0 -j MASQUERADE
```

O mascaramento dos IPs da rede 192.168.10.0/24 se faz necessário por serem IPs privados e não públicos. Assim, o único IP conhecido para a rede Internet será 200.135.37.65. Os testes realizados são apresentados abaixo.

Testes com PING

PING da VM belgica para a VM alemanha

```
PING 192.168.10.52 (192.168.10.52) 56(84) bytes of data. 64 bytes
from 192.168.10.52: icmp_seq=1 ttl=64 time=2.50 ms 64 bytes
from 192.168.10.52: icmp_seq=2 ttl=64 time=0.528 ms 64 bytes
from 192.168.10.52: icmp_seq=3 ttl=64 time=0.527 ms 64 bytes
from 192.168.10.52: icmp_seq=4 ttl=64 time=0.592 ms 64 bytes
from 192.168.10.52: icmp_seq=5 ttl=64 time=0.470 ms 64 bytes
from 192.168.10.52: icmp_seq=6 ttl=64 time=0.434 ms 64 bytes
from 192.168.10.52: icmp_seq=7 ttl=64 time=0.493 ms
--- 192.168.10.52 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6000ms
rtt min/avg/max/mdev = 0.434/0.793/2.508/0.701 ms
```

PING da VM italia para a VM dinamarca

```
PING 192.168.10.45 (192.168.10.45) 56(84) bytes of data. 64 bytes
from 192.168.10.45: icmp_seq=1 ttl=64 time=7.54 ms 64 bytes
from 192.168.10.45: icmp_seq=2 ttl=64 time=1.08 ms 64 bytes
from 192.168.10.45: icmp_seq=3 ttl=64 time=3.07 ms 64 bytes
from 192.168.10.45: icmp_seq=4 ttl=64 time=1.01 ms 64 bytes
from 192.168.10.45: icmp_seq=5 ttl=64 time=0.831 ms 64 bytes
from 192.168.10.45: icmp_seq=6 ttl=64 time=0.843 ms 64 bytes
from 192.168.10.45: icmp_seq=7 ttl=64 time=0.810 ms 64 bytes
from 192.168.10.45: icmp_seq=8 ttl=64 time=0.892 ms
--- 192.168.10.45 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7004ms
rtt min/avg/max/mdev = 0.810/2.011/7.544/2.209 ms
```

PING da VM luxemburgo para a VM escocia

```
PING 192.168.10.55 (192.168.10.55) 56(84) bytes of data. 64 bytes
from 192.168.10.55: icmp_seq=1 ttl=64 time=3.35 ms 64 bytes
from 192.168.10.55: icmp_seq=2 ttl=64 time=0.607 ms 64 bytes
from 192.168.10.55: icmp_seq=3 ttl=64 time=0.649 ms 64 bytes
from 192.168.10.55: icmp_seq=4 ttl=64 time=0.619 ms 64 bytes
from 192.168.10.55: icmp_seq=5 ttl=64 time=0.630 ms
--- 192.168.10.55 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4003ms
rtt min/avg/max/mdev = 0.607/1.171/3.351/1.090 ms
```

PING da VM noruega para a VM grecia

```
PING 192.168.10.51 (192.168.10.51) 56(84) bytes of data. 64 bytes
from 192.168.10.51: icmp_seq=1 ttl=64 time=0.872 ms 64 bytes
from 192.168.10.51: icmp_seq=2 ttl=64 time=1.10 ms 64 bytes
from 192.168.10.51: icmp_seq=3 ttl=64 time=1.02 ms 64 bytes
from 192.168.10.51: icmp_seq=4 ttl=64 time=0.900 ms 64 bytes
from 192.168.10.51: icmp_seq=5 ttl=64 time=0.918 ms 64 bytes
from 192.168.10.51: icmp_seq=6 ttl=64 time=1.12 ms 64 bytes
from 192.168.10.51: icmp_seq=7 ttl=64 time=3.83 ms
--- 192.168.10.51 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6007ms
rtt min/avg/max/mdev = 0.872/1.396/3.836/1.001 ms
```

PING da VM portugal para a VM espanha

```
PING 192.168.10.50 (192.168.10.50) 56(84) bytes of data. 64 bytes
from 192.168.10.50: icmp_seq=1 ttl=64 time=7.01 ms 64 bytes
from 192.168.10.50: icmp_seq=2 ttl=64 time=5.05 ms 64 bytes
from 192.168.10.50: icmp_seq=3 ttl=64 time=2.95 ms 64 bytes
from 192.168.10.50: icmp_seq=4 ttl=64 time=5.01 ms 64 bytes
from 192.168.10.50: icmp_seq=5 ttl=64 time=2.14 ms 64 bytes
```

```
from 192.168.10.50: icmp_seq=6 ttl=64 time=3.49 ms 64 bytes
from 192.168.10.50: icmp_seq=7 ttl=64 time=3.08 ms 64 bytes
from 192.168.10.50: icmp_seq=8 ttl=64 time=2.12 ms
--- 192.168.10.50 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7008ms
rtt min/avg/max/mdev = 2.125/3.861/7.010/1.591 ms
```

PING da VM suecia para a VM irlandia

```
PING 192.168.10.54 (192.168.10.54) 56(84) bytes of data. 64 bytes
from 192.168.10.54: icmp_seq=1 ttl=64 time=7.25 ms 64 bytes
from 192.168.10.54: icmp_seq=2 ttl=64 time=0.798 ms 64 bytes
from 192.168.10.54: icmp_seq=3 ttl=64 time=0.735 ms 64 bytes
from 192.168.10.54: icmp_seq=4 ttl=64 time=0.764 ms 64 bytes
from 192.168.10.54: icmp_seq=5 ttl=64 time=0.669 ms 64 bytes
from 192.168.10.54: icmp_seq=6 ttl=64 time=0.708 ms 64 bytes
from 192.168.10.54: icmp_seq=7 ttl=64 time=0.796 ms

--- 192.168.10.54 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6001ms
rtt min/avg/max/mdev = 0.669/1.675/7.259/2.280 ms
```

PING da VM suica para a VM escocia

```
PING 192.168.10.55 (192.168.10.55) 56(84) bytes of data. 64 bytes
from 192.168.10.55: icmp_seq=1 ttl=64 time=0.765 ms 64 bytes
from 192.168.10.55: icmp_seq=2 ttl=64 time=0.653 ms 64 bytes
from 192.168.10.55: icmp_seq=3 ttl=64 time=0.642 ms 64 bytes
from 192.168.10.55: icmp_seq=4 ttl=64 time=0.565 ms 64 bytes
from 192.168.10.55: icmp_seq=5 ttl=64 time=0.662 ms 64 bytes
from 192.168.10.55: icmp_seq=6 ttl=64 time=0.658 ms

--- 192.168.10.55 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5005ms
rtt min/avg/max/mdev = 0.565/0.657/0.765/0.063 ms
```

7.7.1 Traçando Rotas com o traceroute

Traceroute VM espanha para Servidor hendrix

```
1 192.168.10.1 (192.168.10.1) 5.528 ms 3.121 ms 2.364 ms
2 192.168.1.254 (192.168.1.254) 2.149 ms 4.704 ms 3.092 ms
3 200.135.37.65 (200.135.37.65) 3.064 ms 3.088 ms 3.090 ms
```

Traceroute VM inglaterra para 200.176.2.28

```
1 192.168.10.1 (192.168.10.1) 0.884 ms 0.689 ms 0.500 ms
2 192.168.1.254 (192.168.1.254) 0.765 ms 0.642 ms 0.659 ms
3 200.135.233.254 (200.135.233.254) 0.999 ms 1.019 ms 0.895 ms
4 200.237.201.85 (200.237.201.85) 0.939 ms 0.876 ms 0.810 ms
5 200.237.201.62 (200.237.201.62) 1.147 ms 0.949 ms 0.932 ms
6 200.237.194.251 (200.237.194.251) 1.786 ms 1.393 ms 1.176 ms
7 200.143.252.1 (200.143.252.1) 17.071 ms 6.395 ms 7.250 ms
8 200.143.252.10 (200.143.252.10) 31.227 ms 12.810 ms 19.378 ms
9 200.219.130.38 (200.219.130.38) 40.244 ms 45.652 ms 36.667 ms
10 200.177.169.130 (200.177.169.130) 54.265 ms 52.169 ms 52.474 ms
11 200.176.8.177 (200.176.8.177) 48.370 ms 49.376 ms 51.887 ms
12 200.176.8.254 (200.176.8.254) 49.688 ms 44.147 ms 43.853 ms
13 200.176.2.28 (200.176.2.28) 46.254 ms 51.671 ms 49.672 ms
```

Traceroute VM alemanha para o Google

```
1 192.168.10.1 (192.168.10.1) 0.975 ms 0.527 ms 0.407 ms
2 192.168.1.254 (192.168.1.254) 0.678 ms 7.359 ms 0.750 ms
3 200.135.233.254 (200.135.233.254) 4.635 ms 0.983 ms 0.860 ms
4 200.237.201.85 (200.237.201.85) 0.908 ms 0.717 ms 0.812 ms
5 200.237.201.62 (200.237.201.62) 0.957 ms 0.886 ms 0.878 ms
6 200.237.194.251 (200.237.194.251) 1.523 ms 1.211 ms 1.495 ms
7 200.143.252.1 (200.143.252.1) 6.019 ms 6.655 ms 15.145 ms
```

```
8 200.143.252.10 (200.143.252.10) 22.058 ms 11.704 ms 11.756 ms
9 200.136.34.29 (200.136.34.29) 12.954 ms 12.596 ms 12.514 ms
10 209.85.250.242 (209.85.250.242) 12.803 ms 12.787 ms 13.125 ms
11 209.85.251.210 (209.85.251.210) 24.751 ms 209.85.251.214
(209.85.251.214) 26.126 ms 209.85.251.210 (209.85.251.210) 13.205 ms
12 209.85.193.147 (209.85.193.147) 13.138 ms 13.218 ms 13.491 ms
```

Traceroute VM grecia para a UFSC

```
1 192.168.10.1 (192.168.10.1) 0.781 ms 0.536 ms 0.367 ms
2 192.168.1.254 (192.168.1.254) 0.629 ms 0.559 ms 0.685 ms
3 200.135.233.254 (200.135.233.254) 0.921 ms 0.889 ms 0.898 ms
4 200.237.201.85 (200.237.201.85) 0.537 ms 1.066 ms 0.876 ms
5 200.237.201.62 (200.237.201.62) 0.888 ms 0.954 ms 0.911 ms
6 200.237.194.198 (200.237.194.198) 1.032 ms 0.970 ms 0.958 ms
7 150.162.1.150 (150.162.1.150) 0.887 ms 1.190 ms 0.920 ms
```

7.7.2 Capturando Pacotes com tcpdump

Captura de pacote na interface eth0 da VM dinamarca

```
IP 192.168.1.98.1032 > 239.255.255.250.1900: UDP, length 133 arp
who-has 192.168.1.98 (04:40:04:40:00:48) tell 192.168.1.35 arp
who-has 192.168.1.98 tell 192.168.1.83 IP 192.168.1.71.1088 >
234.8.7.1.1088: UDP, length 64 IP 192.168.10.48.5353 >
224.0.0.251.5353: 0 [3q] [5n] [|domain] IP 192.168.1.10.5353 >
224.0.0.251.5353: 0*- [0q] 1/0/0 A 192.168.1.10 (43) IP
192.168.10.56.5353 > 224.0.0.251.5353: 0*- [0q] 1/0/0 (112) arp
who-has 192.168.1.3 tell 192.168.1.98 IP 192.168.10.43.5353 >
224.0.0.251.5353: 0 [3q] [5n] [|domain] IP 192.168.1.10.5353 >
224.0.0.251.5353: 0*- [0q] 1/0/0 A 192.168.1.10 (43) Captura de
pacote na interface eth0 da VM noruega arp who-has 192.168.1.35 tell
192.168.1.27 IP 0.0.0.0.68 > 255.255.255.255.67: BOOTP/DHCP, Request
```

```

from 00:1a:4d:a5:13:ae, length 300 arp who-has 192.168.1.237 tell
192.168.1.1 IP 0.0.0.0.68 > 255.255.255.255.67: BOOTP/DHCP, Request
from 00:1a:4d:a5:13:ae, length 304 IP 192.168.1.71.1088 >
234.8.7.1.1088: UDP, length 64 arp who-has 192.168.1.237 tell
192.168.1.237 IP 192.168.1.27.138 > 192.168.1.255.138: NBT UDP
PACKET(138) arp who-has 192.168.1.237 tell 192.168.1.237 IP6
fe80::21a:4dff:fea5:1334 > ff02::2: ICMP6, router solicitation,
length 16 IP 192.168.1.71.1088 > 234.8.7.1.1088: UDP, length 64

```

Captura de pacote na interface eth0 da VM suecia

```

IP 172.18.0.4.631 > 172.18.0.255.631: UDP, length 148 IP
192.168.1.4.631 > 192.168.1.255.631: UDP, length 148 IP
192.168.10.46.5353 > 224.0.0.251.5353: 0 [3q] [5n] [|domain] IP
192.168.1.10.5353 > 224.0.0.251.5353: 0*- [0q] 1/0/0 A 192.168.1.10
(43) IP 192.168.10.47.5353 > 224.0.0.251.5353: 0*- [0q] 1/0/0 (110)
arp who-has 192.168.1.121 tell 192.168.1.254 IP 192.168.1.222.5353 >
224.0.0.251.5353: 0 [4q] A (QM)? silas.local. AAAA (QM)?
silas.local.[|domain] IP 192.168.1.41.5353 > 224.0.0.251.5353: 0*-
[0q] 4/0/0[|domain] IP 192.168.1.222.5353 > 224.0.0.251.5353: 0
[54q] [|domain] IP 192.168.1.52.5353 > 224.0.0.251.5353: 0*- [0q]
4/0/0 (Cache flush) AAAA fe80::211:d8ff:fed8:ece0,[|domain]

```

Captura de pacote na interface eth0 da VM suica

```

IP 192.168.1.222.5353 > 224.0.0.251.5353: 0 [2q] A (QM)?
Gismonte.local.[|domain] IP 192.168.1.172.138 > 192.168.1.255.138:
NBT UDP PACKET(138) IP 192.168.10.49.5353 > 224.0.0.251.5353: 0 [3q]
[5n] [|domain] IP 192.168.1.10.5353 > 224.0.0.251.5353: 0*- [0q]
1/0/0 A 192.168.1.10 (43) IP 192.168.10.47.5353 > 224.0.0.251.5353:
0*- [0q] 1/0/0 (110) IP 192.168.1.222.5353 > 224.0.0.251.5353: 0
[3q] AAAA (QM)? irlanda.local. A (QM)? irlanda.local.[|domain] IP
192.168.10.54.5353
> 224.0.0.251.5353: 0*- [0q] 3/0/0[|domain] IP 192.168.1.222.5353 >
224.0.0.251.5353: 0 [2q] A (QM)? Gismonte.local.[|domain]

```


7.8 Configuração de uma Malha de Rede Virtual com Acesso a Internet no Laboratório de Redes I

Objetivo:

- Criar uma malha de rede virtual com acesso à Internet no laboratório de redes I usando a infra-estrutura do mesmo como mostrado na figura 29. Não será alterada a infra-estrutura nem as configurações da rede do laboratório .

Configuração da Malha de Rede Virtual:

- O Laboratório de Redes I possui 17 máquinas reais conectadas a um *switch* de 24 portas, que por sua vez está conectado a rede do CEFET/SC - Unidade São José. Cada máquina real possui configurada uma máquina virtual com sistema operacional Linux Mandriva 2008 instalado. Como as interfaces de rede das máquinas virtuais estão configuradas como *bridged* (conectadas diretamente a rede física) serão criadas quatro redes distintas (com dois clientes cada) interligadas entre si e à rede do CEFET/SC - Unidade São José por oito roteadores.

Atividades:

- Configurar as interfaces de rede das máquinas virtuais já instaladas nas máquinas reais, sendo:
 - Oito máquinas virtuais para serem roteadores;
 - Oito máquinas virtuais para serem clientes.
- Criar uma interface de rede virtual na máquina “mateus” (micro do professor) para prover acesso a Internet.

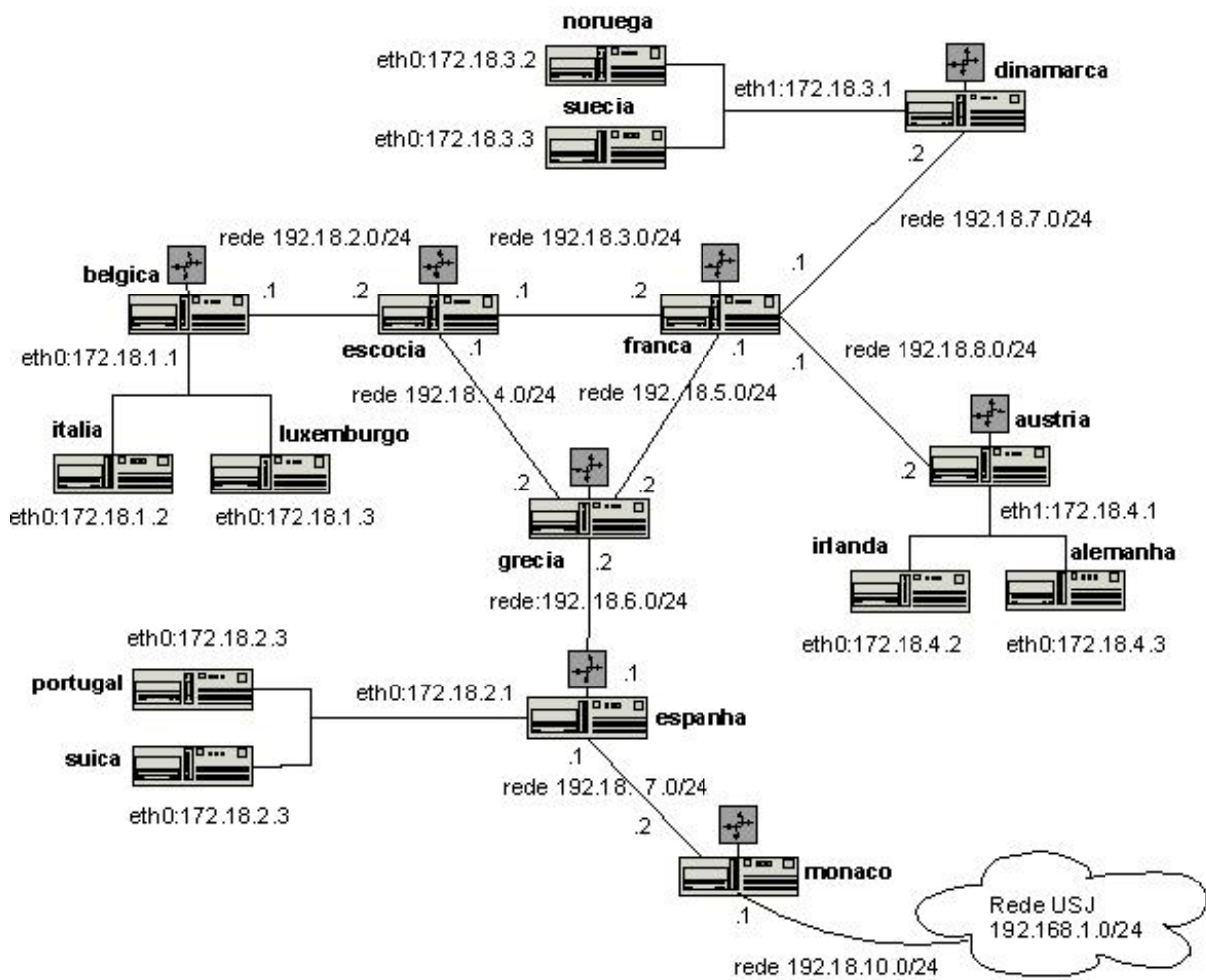


Figura 29: Diagrama da Malha de Rede Virtual com Acesso a Internet no Laboratório de Redes I.

Passo 01 - Máquinas Roteadoras

Configuração das interfaces de rede:

1. Digitar `ifconfig` para listar as interfaces existentes.

Exemplo de `ifconfig`:

```
eth1      Link encap:Ethernet  Endereço de HW 00:0C:29:70:F9:DD
          inet end.: 192.168.10.40  Bcast:192.168.10.255  Masc:255.255.255.0
          endereço inet6: fe80::20c:29ff:fe70:f9dd/64  Escopo:Link
          UP BROADCASTRUNNING MULTICAST  MTU:1500  Métrica:1
          RX packets:2809  errors:0  dropped:0  overruns:0  frame:0
          TX packets:170  errors:0  dropped:0  overruns:0  carrier:0
```

```

colisões:0 txqueuelen:1000
RX bytes:262266 (256.1 KiB) TX bytes:33233 (32.4 KiB)
IRQ:17 Endereço de E/S:0x1400

lo      Link encap:Loopback Local
        inet end.: 127.0.0.1  Masc:255.0.0.0
        endereço inet6: ::1/128 Escopo:Mãquina
        UP LOOPBACKRUNNING  MTU:16436  Métrica:1
        RX packets:2 errors:0 dropped:0 overruns:0 frame:0
        TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
        colisões:0 txqueuelen:0
        RX bytes:144 (144.0 b) TX bytes:144 (144.0 b)

```

2. Configurar o arquivo `network`, digitando na linha de comando:

`vi /etc/sysconfig/network` onde serão definidos os nomes das máquinas, conforme dados da tabela 8. Este arquivo tem os seguintes parâmetros:

- `NETWORKING` = trabalha ou não em rede;
- `GATEWAY` = roteador padrão;
- `GATEWAYDEV` = interface de acesso ao roteador padrão;
- `HOSTNAME` = nome da máquina.

	Router1	Router2	Router3	Router4	Router5	Router6	Router7	Router8
NETWORKING	yes	yes	yes	yes	yes	yes	yes	yes
HOSTNAME	belgica	escocia	franca	dinamarca	austria	grecia	espanha	monaco

Tabela 8: Configuração do arquivo `network`.

Obs.: Para este exemplo não é necessário editar o parâmetro `GATEWAYDEV`; o parâmetro `GATEWAY` será editado a seguir.

3. Configurar o arquivo `ifcfg-ethX`, onde X é o número da interface, conforme dados da tabela 9, digitando na linha de comando:

```
vi /etc/sysconfig/network-scripts/ifcfg-ethX
```

Este arquivo tem os seguintes parâmetros:

- `DEVICE` = nome do dispositivo;

- BOOTPROTO = configuração estática ou dinâmica;
- IPADDR = endereço IP;
- NETMASK = máscara de rede;
- BROADCAST = endereço de broadcast;
- GATEWAY = roteador padrão;
- ONBOOT = interface inicializa no boot da máquina;
- MSDNS1 = endereço do servidor DNS primário.
- MSDNS2 = endereço do servidor DNS secundário.

	belgica	escocia	franca	dinamarca
DEVICE	eth0	eth2	eth2	eth1
BOOTPROTO	static	static	static	static
IPADDR	172.18.1.1	192.18.2.2	192.18.5.1	172.18.3.1
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
BROADCAST	172.18.1.255	192.18.2.255	192.18.5.255	172.18.3.255
GATEWAY	192.18.2.2	192.18.2.2	192.18.5.1	192.18.7.1
ONBOOT	yes	yes	yes	yes
	austria	grecia	espanha	monaco
DEVICE	eth1	eth1	eth0	eth1
BOOTPROTO	static	static	static	static
IPADDR	172.18.4.1	192.18.6.2	192.18.6.1	192.18.10.1
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
BROADCAST	172.18.4.255	192.18.6.255	192.18.6.255	192.18.10.255
GATEWAY	192.18.8.1	192.18.6.2	192.18.6.1	192.18.10.2
ONBOOT	yes	yes	yes	yes

Tabela 9: Configuração do arquivo `ifcfg-ethX`.

Obs.: Para este exemplo não é necessário editar os parâmetros `MSDNS1` e `MSDNS2`.

4. Criar o arquivo `ifcfg-ethX:Y` para criar as interfaces virtuais (onde X é o número da interface real e Y é outro número, a partir de zero, para identificar as interfaces virtuais), conforme dados da tabela 10. Digitar na linha de comando:

`vi/etc/sysconfig/network-scripts/ifcfg-ethX:Y` e no editor `vi` digitar:

- IPADDR = endereço IP;
- NETMASK = máscara de rede.

Obs.: Após estas configurações reiniciar o serviço de rede com o comando: `service network restart`

	belgica	escocia	franca	dinamarca
IPADDR	192.18.2.1 eth0:1	192.18.3.1 eth2:1	192.18.7.1 eth2:1	192.18.7.2 eth1:1
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
IPADDR		192.18.4.1 eth2:2	192.18.8.1 eth2:2	
NETMASK		255.255.255.0	255.255.255.0	
IPADDR			192.18.5.1 eth2:3	
NETMASK			255.255.255.0	
	austria	grecia	espanha	monaco
IPADDR	192.18.8.2 eth1:1	192.18.5.2 eth1:1	192.18.9.1 eth0:1	192.18.9.2 eth1:1
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
IPADDR		192.18.4.2 eth1:2	172.18.2.1 eth1:2	
NETMASK		255.255.255.0	255.255.255.0	

Tabela 10: Configuração do arquivo `ifcfg-ethX:Y`

Em nosso exemplo, todas as máquinas virtuais vão assumir a interface de rede física como sendo sua interface de rede. Como para os roteadores se faz necessário mais de uma interface de rede foram criadas as interfaces virtuais.

Configurando os roteadores:

1. Digitar o comando:

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

Desta forma estamos mudando o bit `ipforward` para 1 e tornando as máquinas roteadores.

2. Adicionando rotas: Com o comando `route -n` é impresso na tela a tabela de roteamento atual.

roteador austria

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usos	Iface
192.18.8.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
172.18.4.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
0.0.0.0	192.18.8.1	0.0.0.0	UG	10	0	0	eth1

roteador belgica

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Uso	Iface
192.18.2.0	0.0.0.0	255.255.255.0	U	10	0	0	eth0
172.18.1.0	0.0.0.0	255.255.255.0	U	10	0	0	eth0
0.0.0.0	192.18.2.2	0.0.0.0	UG	10	0	0	eth0

roteador espanha

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Uso	Iface
192.18.9.0	0.0.0.0	255.255.255.0	U	10	0	0	eth0
172.18.2.0	0.0.0.0	255.255.255.0	U	10	0	0	eth0
192.18.6.0	0.0.0.0	255.255.255.0	U	10	0	0	eth0
0.0.0.0	192.18.2.2	0.0.0.0	UG	10	0	0	eth0

roteador dinamarca

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Uso	Iface
192.18.7.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
172.18.3.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
0.0.0.0	192.18.7.1	0.0.0.0	UG	10	0	0	eth1

roteador franca

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Uso	Iface
---------	----------	-------------	--------	---------	-----	-----	-------

```

192.18.8.0 0.0.0.0 255.255.255.0 U      10      0      0      eth2
192.18.3.0 0.0.0.0 255.255.255.0 U      10      0      0      eth2
192.18.5.0 0.0.0.0 255.255.255.0 U      10      0      0      eth2
192.18.7.0 0.0.0.0 255.255.255.0 U      10      0      0      eth2
0.0.0.0   192.18.5.1 0.0.0.0      UG     10      0      0      eth2

```

roteador escocia

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usos	Interface
192.18.2.0	0.0.0.0	255.255.255.0	U	10	0	0	eth2
192.18.3.0	0.0.0.0	255.255.255.0	U	10	0	0	eth2
192.18.4.0	0.0.0.0	255.255.255.0	U	10	0	0	eth2
0.0.0.0	192.18.2.2	0.0.0.0	UG	10	0	0	eth2

roteador grecia

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usos	Interface
192.18.4.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.18.5.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
192.18.6.0	0.0.0.0	255.255.255.0	U	10	0	0	eth1
0.0.0.0	192.18.6.2	0.0.0.0	UG	10	0	0	eth1

roteador monaco

Tabela de Roteamento IP do Kernel

Destino	Roteador	MáscaraGen.	Opções	Métrica	Ref	Usos	Interface
192.18.9.0	0.0.0.0	255.255.255.0	U	10	0	0	eth0
192.18.10.0	0.0.0.0	255.255.255.0	U	10	0	0	eth0
0.0.0.0	192.18.10.2	0.0.0.0	UG	10	0	0	eth0

Após todas as configurações de rede descritas acima estas serão as tabelas de roteamento padrão para cada roteador. Para adicionar rotas digitamos: `route add -net 192.168.Y.0/24 gw 192.168.Y.Z` onde o primeiro endereço é o da rede que queremos atingir e o segundo endereço é o do roteador que selecionará a melhor rota para alcançar este objetivo. Em nosso exemplo não será necessário adição de rotas.

Passo 02 - Máquinas Clientes

Configuração das interfaces de rede:

3. Configurar o arquivo `network`, digitando na linha de comando:

`vi /etc/sysconfig/network` onde serão definidos os nomes das máquinas, conforme dados da tabela 11. Este arquivo tem os seguintes parâmetros:

- NETWORKING = trabalha ou não em rede;
- GATEWAY = roteador padrão;
- GATEWAYDEV = interface de acesso ao roteador padrão;
- HOSTNAME = nome da máquina;

	maq1	maq2	maq3	maq4	maq5	maq6	maq7	maq8
NETWORKING	yes	yes	yes	yes	yes	yes	yes	yes
HOSTNAME	italia	luxemburgo	inglaterra	suecia	irlanda	alemanha	portugal	suica

Tabela 11: Configuração do arquivo `network`.

Obs.: Para este exemplo não é necessário editar o parâmetro `GATEWAYDEV`; o parâmetro `GATEWAY` será editado no próximo arquivo.

4. Configurar o arquivo `ifcfg-ethX`, onde X é o número da interface, conforme dados da tabela 12, digitando na linha de comando:

`vi /etc/sysconfig/network-scripts/ifcfg-ethX`

Este arquivo tem os seguintes parâmetros:

- **DEVICE** = nome do dispositivo;
- **BOOTPROTO** = configuração estática ou dinâmica;
- **IPADDR** = endereço IP;
- **NETMASK** = máscara de rede;
- **BROADCAST** = endereço de broadcast;
- **GATEWAY** = roteador padrão;
- **ONBOOT** = interface inicializa no boot da máquina;
- **MSDNS1** = endereço do servidor DNS primário;
- **MSDNS2** = endereço do servidor DNS secundário.

	italia	luxemburgo	inglaterra	suecia
DEVICE	eth0	eth0	eth0	eth0
BOOTPROTO	static	static	static	static
IPADDR	172.18.1.2	172.18.1.3	172.18.3.2	172.18.3.3
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
BROADCAST	172.18.1.255	172.18.1.255	172.18.3.255	172.18.3.255
GATEWAY	172.18.1.1	172.18.1.1	172.18.3.1	172.18.3.1
ONBOOT	yes	yes	yes	yes
	irlanda	alemanha	portugal	suica
DEVICE	eth0	eth0	eth0	eth0
BOOTPROTO	static	static	static	static
IPADDR	172.18.4.2	172.18.4.3	172.18.2.2	172.18.2.3
NETMASK	255.255.255.0	255.255.255.0	255.255.255.0	255.255.255.0
BROADCAST	172.18.4.255	172.18.4.255	172.18.2.255	172.18.2.255
GATEWAY	172.18.4.1	172.18.4.1	172.18.2.1	172.18.2.1
ONBOOT	yes	yes	yes	yes

Tabela 12: Configuração do arquivo `ifcfg-ethX`.

Observações:

- Para este exemplo não é necessário editar os parâmetros `MSDNS1` e `MSDNS2`;
- Após estas configurações reiniciar o serviço de rede: `service network restart`.

Passo 03 - Criar uma interface de rede virtual na máquina mateus

1. Criar uma interface de rede virtual na máquina “mateus”.

Criar o arquivo `ifcfg-ethX:Y` para criar as interfaces virtuais, (onde X é o número da interface real e Y é outro número, a partir de zero, para identificar as interfaces virtuais). Digitar na linha de comando:

`vi/etc/sysconfig/network-scripts/ifcfg-ethX:Y` e no editor `vi` digitar:

- `IPADDR` = endereço IP;
- `NETMASK` = máscara de rede.

Na máquina real **mateus** a interface virtual terá a configuração:

- `IPADDR` = 192.168.10.2;
- `NETMASK` = 255.255.255.0.

Obs.: Após esta configuração reiniciar o serviço de rede com o comando: `service network restart`

2. Configurar a máquina real “mateus” como roteador para rede 192.168.10.0/24

Digitar o comando:

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

Desta forma estamos mudando o bit `ipforward` para 1 e tornando as máquinas roteadores.

Para adicionar rotas digitamos:

```
route add -net 192.168.Y.0/24 gw 192.168.Y.Z
```

onde o primeiro endereço é o da rede que queremos atingir e o segundo endereço é o do roteador que selecionará a melhor rota para alcançar este objetivo:

```
route add -net 192.168.10.0/24 gw 192.168.10.1
```

3. Configurar a máquina real “mateus” para fazer mascaramento da rede 192.168.10.0/24:

```
iptables -t nat -A POSTROUTING -s 192.168.10.0/24 -o eth0 -j MASQUERADE
```

O mascaramento dos IPs da rede 192.168.10.0/24 se faz necessário por serem IPs privados e não públicos. Assim, o único IP conhecido para a rede Internet será 200.135.37.65

Redes entre os roteadores:

- 192.18.2.0/24 b elgica - esc ocia
- 192.18.3.0/24 escocia - franca
- 192.18.4.0/24 escocia - grecia
- 192.18.5.0/24 franca - grecia
- 192.18.6.0/24 grecia - espanha
- 192.18.7.0/24 franca - dinamarca
- 192.18.8.0/24 franca - austria
- 192.18.9.0/24 espaha - monaco
- 192.18.10.0/24 monaco - mateus (192.168.1.0/24)

Os testes realizados s o apresentados abaixo.

7.8.1 Testes com PING

PING da VM inglaterra para a VM portugal

```
PING 172.18.2.2 (172.18.2.2) 56(84) bytes of data.  
From 192.18.2.2: icmp_seq=1 Redirect Host(New nexthop: 172.18.2.2)  
64 bytes  
from 172.18.2.2: icmp_seq=1 ttl=64 time=2.76 ms 64 bytes  
from 172.18.2.2: icmp_seq=2 ttl=64 time=1.94 ms  
From 192.18.2.2: icmp_seq=3 Redirect Host(New nexthop: 172.18.2.2)  
64 bytes  
from 172.18.2.2: icmp_seq=3 ttl=64 time=1.89 ms 64 bytes  
from 172.18.2.2: icmp_seq=4 ttl=64 time=1.78 ms 64 bytes  
from 172.18.2.2: icmp_seq=5 ttl=64 time=1.83 ms  
from 192.18.2.2: icmp_seq=6 Redirect Host(New nexthop: 172.18.2.2)
```

64 bytes

from 172.18.2.2: icmp_seq=6 ttl=64 time=1.65 ms 64 bytes

from 172.18.2.2: icmp_seq=7 ttl=64 time=1.79 ms

--- 172.18.2.2 ping statistics ---

7 packets transmitted, 7 received, 0% packet loss, time 6005ms

rtt min/avg/max/mdev = 1.650/1.952/2.764/0.343 ms

PING da VM inglaterra para a VM itália

PING 172.18.1.2 (172.18.1.2) 56(84) bytes of data.

From 192.18.2.2: icmp_seq=1 Redirect Host(New nexthop: 172.18.1.2)

64 bytes from 172.18.1.2: icmp_seq=1 ttl=64 time=1.17 ms From

192.18.2.2: icmp_seq=2 Redirect Host(New nexthop: 172.18.1.2) 64

bytes from 172.18.1.2: icmp_seq=2 ttl=64 time=1.09 ms From

192.18.2.2: icmp_seq=3 Redirect Host(New nexthop: 172.18.1.2) 64

bytes from 172.18.1.2: icmp_seq=3 ttl=64 time=1.04 ms From

192.18.2.2: icmp_seq=4 Redirect Host(New nexthop: 172.18.1.2) 64

bytes from 172.18.1.2: icmp_seq=4 ttl=64 time=1.05 ms From

192.18.2.2: icmp_seq=5 Redirect Host(New nexthop: 172.18.1.2) 64

bytes from 172.18.1.2: icmp_seq=5 ttl=64 time=0.943 ms

--- 172.18.1.2 ping statistics ---

5 packets transmitted, 5 received, 0% packet loss, time 4003ms

rtt min/avg/max/mdev = 0.943/1.060/1.170/0.079 ms

PING da VM suecia para a VM itália

PING 172.18.1.2 (172.18.1.2) 56(84) bytes of data. From 192.18.2.2:

icmp_seq=1 Redirect Host(New nexthop: 172.18.1.2) 64 bytes from

172.18.1.2: icmp_seq=1 ttl=64 time=1.45 ms From 192.18.2.2:

icmp_seq=2 Redirect Host(New nexthop: 172.18.1.2) 64 bytes from

172.18.1.2: icmp_seq=2 ttl=64 time=1.22 ms 64 bytes from 172.18.1.2:

icmp_seq=3 ttl=64 time=1.15 ms From 192.18.2.2: icmp_seq=4 Redirect

```
Host(New nexthop: 172.18.1.2) 64 bytes
from 172.18.1.2: icmp_seq=4 ttl=64 time=2.34 ms 64 bytes
from 172.18.1.2: icmp_seq=5 ttl=64 time=1.31 ms 64 bytes
from 172.18.1.2: icmp_seq=6 ttl=64 time=1.35 ms
from 192.18.2.2: icmp_seq=7 Redirect Host(New nexthop: 172.18.1.2)
64 bytes from 172.18.1.2: icmp_seq=7 ttl=64 time=1.19 ms 64 bytes
from 172.18.1.2: icmp_seq=8 ttl=64 time=1.20 ms
```

--- 172.18.1.2 ping statistics ---

```
8 packets transmitted, 8 received, 0% packet loss, time 7005ms
rtt min/avg/max/mdev = 1.153/1.406/2.341/0.366 ms
```

PING da VM suecia para a VM alemanha

```
PING 172.18.4.3 (172.18.4.3) 56(84) bytes of data. 64 bytes
from 172.18.4.3: icmp_seq=1 ttl=62 time=6.03 ms 64 bytes
from 172.18.4.3: icmp_seq=2 ttl=62 time=5.10 ms 64 bytes
from 172.18.4.3: icmp_seq=3 ttl=62 time=2.75 ms 64 bytes
from 172.18.4.3: icmp_seq=4 ttl=62 time=2.72 ms 64 bytes
from 172.18.4.3: icmp_seq=5 ttl=62 time=6.73 ms
```

--- 172.18.4.3 ping statistics ---

```
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 2.723/4.670/6.735/1.661 ms
```

PING da VM suecia para a VM suíça

```
PING 172.18.2.3 (172.18.2.3) 56(84) bytes of data.
From 192.18.6.1: icmp_seq=1 Redirect Host(New nexthop: 172.18.2.3)
64 bytes
from 172.18.2.3: icmp_seq=1 ttl=64 time=3.99 ms 64 bytes
from 172.18.2.3: icmp_seq=2 ttl=64 time=2.75 ms 64 bytes
from 172.18.2.3: icmp_seq=3 ttl=64 time=3.08 ms 64 bytes
from 172.18.2.3: icmp_seq=4 ttl=64 time=9.21 ms 64 bytes
```

from 172.18.2.3: icmp_seq=5 ttl=64 time=6.37 ms

--- 172.18.2.3 ping statistics ---

5 packets transmitted, 5 received, 0% packet loss, time 4009ms
rtt min/avg/max/mdev = 2.753/5.085/9.218/2.423 ms

PING da VM portugal para a VM irlandia

PING 172.18.4.2 (172.18.4.2) 56(84) bytes of data. 64 bytes

from 172.18.4.2: icmp_seq=1 ttl=61 time=4.85 ms 64 bytes

from 172.18.4.2: icmp_seq=2 ttl=61 time=3.38 ms 64 bytes

from 172.18.4.2: icmp_seq=3 ttl=61 time=3.12 ms 64 bytes

from 172.18.4.2: icmp_seq=4 ttl=61 time=3.68 ms 64 bytes

from 172.18.4.2: icmp_seq=5 ttl=61 time=3.29 ms 64 bytes

from 172.18.4.2: icmp_seq=6 ttl=61 time=3.04 ms 64 bytes

from 172.18.4.2: icmp_seq=7 ttl=61 time=3.17 ms

--- 172.18.4.2 ping statistics ---

7 packets transmitted, 7 received, 0% packet loss, time 6004ms
rtt min/avg/max/mdev = 3.043/3.509/4.854/0.584 ms

PING da VM italia para a VM irlandia

PING 172.18.4.2 (172.18.4.2) 56(84) bytes of data. 64 bytes

from 172.18.4.2: icmp_seq=1 ttl=62 time=1.43 ms 64 bytes

from 172.18.4.2: icmp_seq=2 ttl=62 time=1.07 ms 64 bytes

from 172.18.4.2: icmp_seq=3 ttl=62 time=1.22 ms 64 bytes

from 172.18.4.2: icmp_seq=4 ttl=62 time=1.19 ms 64 bytes

from 172.18.4.2: icmp_seq=5 ttl=62 time=1.31 ms 64 bytes

from 172.18.4.2: icmp_seq=6 ttl=62 time=1.31 ms 64 bytes

from 172.18.4.2: icmp_seq=7 ttl=62 time=7.24 ms

--- 172.18.4.2 ping statistics ---

7 packets transmitted, 7 received, 0% packet loss, time 6003ms

rtt min/avg/max/mdev = 1.078/2.116/7.249/2.098 ms

PING da VM (roteador) monaco para a máquina real DK (192.168.1.1)

```
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data. 64 bytes
from 192.168.1.1: icmp_seq=1 ttl=64 time=6.04 ms 64 bytes
from 192.168.1.1: icmp_seq=2 ttl=64 time=27.9 ms 64 bytes
from 192.168.1.1: icmp_seq=3 ttl=64 time=4.36 ms 64 bytes
from 192.168.1.1: icmp_seq=4 ttl=64 time=7.79 ms 64 bytes
from 192.168.1.1: icmp_seq=5 ttl=64 time=3.83 ms 64 bytes
from 192.168.1.1: icmp_seq=6 ttl=64 time=3.94 ms 64 bytes
from 192.168.1.1: icmp_seq=7 ttl=64 time=3.68 ms 64 bytes
from 192.168.1.1: icmp_seq=8 ttl=64 time=4.37 ms 64 bytes
from 192.168.1.1: icmp_seq=9 ttl=64 time=6.50 ms
--- 192.168.1.1 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8226ms
rtt min/avg/max/mdev = 3.689/7.615/27.990/7.326 ms
```

PING da VM (roteador) monaco para a máquina real Hendrix (200.135.37.65)

```
PING 200.135.37.65 (200.135.37.65) 56(84) bytes of data. 64 bytes
from 200.135.37.65: icmp_seq=1 ttl=63 time=5.01 ms 64 bytes
from 200.135.37.65: icmp_seq=2 ttl=63 time=7.65 ms 64 bytes
from 200.135.37.65: icmp_seq=3 ttl=63 time=5.14 ms 64 bytes
from 200.135.37.65: icmp_seq=4 ttl=63 time=9.88 ms 64 bytes
from 200.135.37.65: icmp_seq=5 ttl=63 time=4.54 ms 64 bytes
from 200.135.37.65: icmp_seq=6 ttl=63 time=4.61 ms

--- 200.135.37.65 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5149ms
rtt min/avg/max/mdev = 4.547/6.142/9.882/1.977 ms
```

PING da VM (roteador) mônaco para a VM italia

```
PING 172.18.1.2 (172.18.1.2) 56(84) bytes of data.
From 192.18.9.1: icmp_seq=1 Redirect Host(New nexthop: 172.18.1.2)
64 bytes
from 172.18.1.2: icmp_seq=1 ttl=64 time=27.9 ms
From 192.18.9.1: icmp_seq=2 Redirect Host(New nexthop: 172.18.1.2)
64 bytes
from 172.18.1.2: icmp_seq=2 ttl=64 time=9.60 ms 64 bytes
from 172.18.1.2: icmp_seq=3 ttl=64 time=11.2 ms 64 bytes
from 172.18.1.2: icmp_seq=4 ttl=64 time=4.18 ms 64 bytes
from 172.18.1.2: icmp_seq=5 ttl=64 time=6.51 ms 64 bytes
from 172.18.1.2: icmp_seq=6 ttl=64 time=5.17 ms 64 bytes
from 172.18.1.2: icmp_seq=7 ttl=64 time=5.81 ms 64 bytes
from 172.18.1.2: icmp_seq=8 ttl=64 time=5.81 ms

--- 172.18.1.2 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7243ms
rtt min/avg/max/mdev = 4.189/9.541/27.993/7.315 ms
```

PING da VM (roteador) mônaco para a VM alemanha

```
PING 172.18.4.2 (172.18.4.2) 56(84) bytes of data.
From 192.18.9.1: icmp_seq=1 Redirect Host(New nexthop: 172.18.4.2)
64 bytes
from 172.18.4.2: icmp_seq=1 ttl=60 time=7.68 ms 64 bytes
from 172.18.4.2: icmp_seq=2 ttl=60 time=4.67 ms 64 bytes
from 172.18.4.2: icmp_seq=3 ttl=60 time=4.15 ms 64 bytes
from 172.18.4.2: icmp_seq=4 ttl=60 time=3.35 ms 64 bytes
from 172.18.4.2: icmp_seq=5 ttl=60 time=18.1 ms 64 bytes
from 172.18.4.2: icmp_seq=6 ttl=60 time=4.58 ms

--- 172.18.4.2 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5020ms
rtt min/avg/max/mdev = 3.351/7.092/18.104/5.105 ms
```

PING da VM (roteador) mônaco para a VM portugal


```
PING 172.18.2.2 (172.18.2.2) 56(84) bytes of data. 64 bytes
from 172.18.2.2: icmp_seq=1 ttl=64 time=4.02 ms 64 bytes
from 172.18.2.2: icmp_seq=2 ttl=64 time=1.40 ms 64 bytes
from 172.18.2.2: icmp_seq=3 ttl=64 time=2.06 ms 64 bytes
from 172.18.2.2: icmp_seq=4 ttl=64 time=3.36 ms 64 bytes
from 172.18.2.2: icmp_seq=5 ttl=64 time=1.98 ms 64 bytes
from 172.18.2.2: icmp_seq=6 ttl=64 time=3.59 ms
```

```
--- 172.18.2.2 ping statistics ---
```

```
6 packets transmitted, 6 received, 0% packet loss, time 5008ms
rtt min/avg/max/mdev = 1.403/2.740/4.026/0.968 ms
```

PING da VM (roteador) monaco para a VM suécia

```
PING 172.18.3.3 (172.18.3.3) 56(84) bytes of data. 64 bytes
from 172.18.3.3: icmp_seq=1 ttl=61 time=17.8 ms 64 bytes
from 172.18.3.3: icmp_seq=2 ttl=61 time=8.00 ms 64 bytes
from 172.18.3.3: icmp_seq=3 ttl=61 time=5.11 ms 64 bytes
from 172.18.3.3: icmp_seq=4 ttl=61 time=4.24 ms 64 bytes
from 172.18.3.3: icmp_seq=5 ttl=61 time=4.25 ms 64 bytes
from 172.18.3.3: icmp_seq=6 ttl=61 time=4.07 ms
```

```
--- 172.18.3.3 ping statistics ---
```

```
6 packets transmitted, 6 received, 0% packet loss, time 5010ms
rtt min/avg/max/mdev = 4.075/7.263/17.890/4.940 ms
```

PING da máquina real mateus (192.168.1.56) para o VM (roteador) escócia (eth2:2)

```
PING 192.18.4.1 (192.18.4.1) 56(84)bytes of data. 64 bytes
from 192.18.4.1: icmp_seq=1 ttl=240 time=222 ms 64 bytes
from 192.18.4.1: icmp_seq=2 ttl=240 time=220 ms 64 bytes
from 192.18.4.1: icmp_seq=3 ttl=240 time=198 ms 64 bytes
from 192.18.4.1: icmp_seq=4 ttl=240 time=207 ms 64 bytes
```

```
from 192.18.4.1: icmp_seq=5 ttl=240 time=223 ms
```

```
--- 192.18.4.1 ping statistics ---
```

```
6 packets transmitted, 5 received, 16% packet loss, time 5000ms
```

```
rtt min/avg/max/mdev = 198.377/214.455/223.270/9.825 ms
```

PING da máquina real mateus (192.168.1.56) para o VM (roteador) escocia (eth2:1)

```
PING 192.18.3.1 (192.18.3.1) 56(84)bytes of data. 64 bytes
```

```
from 192.18.3.1: icmp_seq=1 ttl=240 time=196 ms 64 bytes
```

```
from 192.18.3.1: icmp_seq=2 ttl=240 time=240 ms 64 bytes
```

```
from 192.18.3.1: icmp_seq=3 ttl=240 time=218 ms 64 bytes
```

```
from 192.18.3.1: icmp_seq=4 ttl=240 time=229 ms 64 bytes
```

```
from 192.18.3.1: icmp_seq=5 ttl=240 time=215 ms
```

```
--- 192.18.3.1 ping statistics ---
```

```
5 packets transmitted, 5 received, 0% packet loss, time 3998ms
```

```
rtt min/avg/max/mdev = 196.181/219.846/240.634/14.856 ms
```

PING da máquina real mateus (192.168.1.56) para o VM (roteador) escocia (eth0:1)

```
PING 192.18.9.1 (192.18.9.1) 56(84) bytes of data.
```

```
From 192.18.10.1: icmp_seq=1 Redirect Host(New nexthop: 192.18.9.1)
```

```
64 bytes from 192.18.9.1: icmp_seq=1 ttl=64 time=7.36 ms From
```

```
192.18.10.1: icmp_seq=2 Redirect Host(New nexthop: 192.18.9.1) 64
```

```
bytes from 192.18.9.1: icmp_seq=2 ttl=64 time=6.48 ms From
```

```
192.18.10.1: icmp_seq=3 Redirect Host(New nexthop: 192.18.9.1) 64
```

```
bytes from 192.18.9.1: icmp_seq=3 ttl=64 time=7.63 ms From
```

```
192.18.10.1: icmp_seq=4 Redirect Host(New nexthop: 192.18.9.1) 64
```

```
bytes from 192.18.9.1: icmp_seq=4 ttl=64 time=1.45 ms From
```

```
192.18.10.1: icmp_seq=5 Redirect Host(New nexthop: 192.18.9.1) 64
```

```
bytes from 192.18.9.1: icmp_seq=5 ttl=64 time=2.24 ms
```

```
--- 192.18.9.1 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 4000ms  
rtt min/avg/max/mdev = 1.458/5.036/7.632/2.641 ms
```

PING da máquina real mateus (192.168.1.56) para o VM (roteador) escocia (eth0:2)

```
PING 192.18.6.1 (192.18.6.1) 56(84) bytes of data. 64 bytes  
from 192.18.6.1: icmp_seq=1 ttl=48 time=222 ms 64 bytes  
from 192.18.6.1: icmp_seq=2 ttl=48 time=244 ms 64 bytes  
from 192.18.6.1: icmp_seq=3 ttl=48 time=217 ms 64 bytes  
from 192.18.6.1: icmp_seq=4 ttl=48 time=205 ms 64 bytes  
from 192.18.6.1: icmp_seq=5 ttl=48 time=222 ms 64 bytes  
from 192.18.6.1: icmp_seq=6 ttl=48 time=260 ms
```

```
--- 192.18.6.1 ping statistics ---  
7 packets transmitted, 6 received, 14% packet loss, time 5999ms  
rtt min/avg/max/mdev = 205.264/228.742/260.377/18.396 ms
```

PING da VM (roteador) monaco para a VM (roteador) franca

```
PING 192.18.5.1 (192.18.5.1) 56(84) bytes of data. 64 bytes  
from 192.18.5.1: icmp_seq=1 ttl=64 time=27.4 ms 64 bytes  
from 192.18.5.1: icmp_seq=2 ttl=64 time=6.45 ms 64 bytes  
from 192.18.5.1: icmp_seq=3 ttl=64 time=7.26 ms 64 bytes  
from 192.18.5.1: icmp_seq=4 ttl=64 time=5.40 ms 64 bytes  
from 192.18.5.1: icmp_seq=5 ttl=64 time=7.61 ms
```

```
--- 192.18.5.1 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 4274ms  
rtt min/avg/max/mdev = 5.409/10.839/27.449/8.339 ms
```

PING da VM (roteador) grecia para a VM (roteador) escócia

```
PING 172.18.4.1 (172.18.4.1) 56(84) bytes of data.
From 192.18.5.1: icmp_seq=1 Redirect Host(New nexthop: 192.18.8.2)
64 bytes
from 172.18.4.1: icmp_seq=1 ttl=63 time=6.94 ms 64 bytes
from 172.18.4.1: icmp_seq=2 ttl=63 time=1.24 ms 64 bytes
from 172.18.4.1: icmp_seq=3 ttl=63 time=1.14 ms 64 bytes
from 172.18.4.1: icmp_seq=4 ttl=63 time=1.31 ms 64 bytes
from 172.18.4.1: icmp_seq=5 ttl=63 time=1.31 ms

--- 172.18.4.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4002ms
rtt min/avg/max/mdev = 1.141/2.391/6.943/2.277 ms
```

PING da VM (roteador) escocia para a VM (roteador) espanha

```
PING 192.18.9.1 (192.18.9.1) 56(84) bytes of data. 64 bytes
from 192.18.9.1: icmp_seq=1 ttl=64 time=1.62 ms 64 bytes
from 192.18.9.1: icmp_seq=2 ttl=64 time=1.34 ms 64 bytes
from 192.18.9.1: icmp_seq=3 ttl=64 time=1.49 ms 64 bytes
from 192.18.9.1: icmp_seq=4 ttl=64 time=1.39 ms 64 bytes
from 192.18.9.1: icmp_seq=5 ttl=64 time=1.46 ms

--- 192.18.9.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4004ms
rtt min/avg/max/mdev = 1.345/1.465/1.623/0.094 ms
```

PING da VM (roteador) dinamarca para a VM (roteador) grécia

```
PING 192.18.6.2 (192.18.6.2) 56(84) bytes of data. 64 bytes
from 192.18.6.2: icmp_seq=1 ttl=64 time=2.09 ms
From 192.18.7.1: icmp_seq=2 Redirect Host(New nexthop: 192.18.6.2)
64 bytes from 192.18.6.2: icmp_seq=2 ttl=64 time=13.4 ms
From 192.18.7.1: icmp_seq=3 Redirect Host(New nexthop: 192.18.6.2)
64 bytes
```

```
from 192.18.6.2: icmp_seq=3 ttl=64 time=13.3 ms 64 bytes
from 192.18.6.2: icmp_seq=4 ttl=64 time=2.29 ms
```

```
--- 192.18.6.2 ping statistics ---
```

```
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
```

```
rtt min/avg/max/mdev = 2.091/7.799/13.445/5.608 ms
```

PING da VM (roteador) franca para a VM (roteador) espanha

```
PING 192.18.6.1 (192.18.6.1) 56(84) bytes of data. 64 bytes
```

```
from 192.18.6.1: icmp_seq=1 ttl=64 time=2.98 ms 64 bytes
```

```
from 192.18.6.1: icmp_seq=2 ttl=64 time=2.21 ms 64 bytes
```

```
from 192.18.6.1: icmp_seq=3 ttl=64 time=1.99 ms 64 bytes
```

```
from 192.18.6.1: icmp_seq=4 ttl=64 time=2.09 ms
```

```
--- 192.18.6.1 ping statistics ---
```

```
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
```

```
rtt min/avg/max/mdev = 1.993/2.319/2.983/0.393 ms
```

7.8.2 Traçando Rotas com o traceroute

Traceroute VM monaco para VM portugal

```
1 192.18.6.1 (192.18.6.1) 4.765 ms 5.311 ms 7.799 ms
```

```
2 172.18.2.2 (172.18.2.2) 7.673 ms 11.949 ms 12.389 ms
```

Traceroute VM suecia para VM suíça

```
1 192.18.7.1 (192.18.7.1) 2.720 ms 13.949 ms 13.492 ms
```

```
2 192.18.5.2 (192.18.5.2) 4.173 ms 20.598 ms 9.319 ms
```

```
3 192.18.6.1 (192.18.6.1) 9.808 ms 16.405 ms 6.013 ms
```

```
4 172.18.2.3 (172.18.2.3) 15.202 ms 7.529 ms 10.404 ms
```

Traceroute VM dinamarca para VM franca

```
1 192.18.2.2 (192.18.2.2) 1.428 ms 1.126 ms 0.889 ms
2 192.18.9.1 (192.18.9.1) 54.393 ms 27.420 ms 25.065 ms
```

Traceroute VM franca para VM suíça

```
1 192.18.5.2 (192.18.5.2) 2.713 ms 0.987 ms 0.977 ms
2 192.18.6.1 (192.18.6.1) 4.587 ms 3.148 ms 1.936 ms
3 172.18.2.3 (172.18.2.3) 5.537 ms 3.366 ms 3.890 ms
```

7.8.3 Capturando Pacotes com tcpdump

Captura de pacote na interface eth1 da VM grécia

```
IP 192.18.5.1 > 192.18.6.1: ICMP echo request, id 32528, seq 177,
length 64 IP 192.18.5.1 > 192.18.6.1: ICMP echo request, id 32528,
seq 177, length 64 IP 192.168.1.2.138 > 192.168.1.255.138: NBT UDP
PACKET(138) arp who-has 192.168.1.156 tell 192.168.1.1 IP
192.168.1.156.138 > 192.168.1.255.138: NBT UDP PACKET(138) IP
192.18.5.1 > 192.18.6.1: ICMP echo request, id 32528, seq 178,
length 64 IP 192.18.5.1 > 192.18.6.1: ICMP echo request, id 32528,
seq 178, length 64 IP 192.18.5.1 > 192.18.6.1: ICMP echo request, id
32528, seq 179, length 64 IP 192.18.5.1 > 192.18.6.1: ICMP echo
request, id 32528, seq 179, length 64 IP 192.18.5.1 > 192.18.6.1:
ICMP echo request, id 32528, seq 180, length 64 IP 192.18.5.1 >
192.18.6.1: ICMP echo request, id 32528, seq 180, length 64 arp
who-has 192.168.1.3 tell 192.168.1.201 IP 192.18.5.1 > 192.18.6.1:
ICMP echo request, id 32528, seq 181, length 64 IP 192.18.5.1 >
192.18.6.1: ICMP echo request, id 32528, seq 181, length 64 IP
172.18.2.3.5353 > 224.0.0.251.5353: 0 [3q] [5n] [|domain]
```

Captura de pacote na interface eth2 da VM escócia IP

```
192.18.2.1.5353 > 224.0.0.251.5353: 0 [2q] [2n][|domain] IP
192.18.2.1.5353 > 224.0.0.251.5353: 0*- [0q] 1/0/0 (Cache flush) A
192.18.12.1 (45) IP 192.168.1.10.5353 > 224.0.0.251.5353: 0*- [0q]
1/0/0 A 192.168.1.10 (43) IP 192.168.10.56.5353 > 224.0.0.251.5353:
0*- [0q] 1/0/0 (112) IP 192.18.2.1.5353 > 224.0.0.251.5353: 0*- [0q]
2/0/0 (Cache flush) PTR[|domain] IP 172.18.0.4.631 >
172.18.0.255.631: UDP, length 146 IP 192.168.1.4.631 >
192.168.1.255.631: UDP, length 146 IP 192.18.2.1.5353 >
224.0.0.251.5353: 0 [2q] [2n] ANY (QM)?
1.12.18.192.in-addr.arpa.[|domain] IP 192.168.1.10.5353 >
224.0.0.251.5353: 0*- [0q] 1/0/0 A 192.168.1.10 (43) IP
192.18.2.1.5353 > 224.0.0.251.5353: 0 [2q] [2n] ANY (QM)?
1.12.18.192.in-addr.arpa.[|domain]
```

Captura de pacote na interface eth2 da VM franca

```
IP 172.18.0.4.631 > 172.18.0.255.631: UDP, length 148 IP
192.168.1.4.631 > 192.168.1.255.631: UDP, length 148 IP
172.18.3.2.5353 > 224.0.0.251.5353: 0 [3q] [5n][|domain] IP
172.18.2.3.5353 > 224.0.0.251.5353: 0*- [0q] 1/0/0 (109) IP
192.168.1.10.5353 > 224.0.0.251.5353: 0*- [0q] 1/0/0 A 192.168.1.10
(43) IPX 00000000.00:04:00:f5:9d:f7.83c2 >
00000000.ff:ff:ff:ff:ff:ff.0452: ipx-sap-resp 055e
'ET000400F59DF7[|ipx 64] (NOV-ETHII) IPX
00000000.00:04:00:f5:9d:f7.83c2 > 00000000.ff:ff:ff:ff:ff:ff.0452:
ipx-sap-resp 055e 'ET000400F59DF7[|ipx 64]
```

Captura de pacote na interface eth0 da VM suíça

```
IP 192.168.1.222.5353 > 224.0.0.251.5353: 0 [2q] A (QM)?
Gismonte.local.[|domain] IP 192.168.1.172.138 > 192.168.1.255.138:
NBT UDP PACKET(138) IP 192.168.10.49.5353 > 224.0.0.251.5353: 0 [3q]
[5n][|domain] IP 192.168.1.10.5353 > 224.0.0.251.5353: 0*- [0q]
1/0/0 A 192.168.1.10 (43) IP 192.168.10.47.5353 > 224.0.0.251.5353:
```

```
0*- [0q] 1/0/0 (110) IP 192.168.1.222.5353 > 224.0.0.251.5353: 0
[3q] AAAA (QM)? irlanda.local. A (QM)? irlanda.local.[|domain] IP
192.168.10.54.5353 > 224.0.0.251.5353: 0*- [0q] 3/0/0[|domain] IP
192.168.1.222.5353 > 224.0.0.251.5353: 0 [2q] A (QM)?
Gismonte.local.[|domain]
```


Referências

- ABRAHAM, J. P. The undergraduate computer networking course, university of texas. In: ACM SIGCOMM, WORKSHOP ON COMPUTER NETWORKING: CURRICULUM DESIGNS AND EDUCATIONAL CHALLENGES. Pittsburgh, PA, 2002.
- ABURDENE, M. An undergraduate networked system laboratory, bucknell university. In: ACM SIGCOMM, WORKSHOP ON COMPUTER NETWORKING: CURRICULUM DESIGNS AND EDUCATIONAL CHALLENGES. Pittsburgh, PA, 2002.
- AL-SHAER, E.; BREWSTER, G. Graduate level curriculum and laboratory courses for computer networking, university chicago. In: ACM SIGCOMM, WORKSHOP ON COMPUTER NETWORKING: CURRICULUM DESIGNS AND EDUCATIONAL CHALLENGES. Pittsburgh, PA, 2002.
- BURROUGHS, A. The laboratory component of a networking course, humboldt state university. In: ACM SIGCOMM, WORKSHOP ON COMPUTER NETWORKING: CURRICULUM DESIGNS AND EDUCATIONAL CHALLENGES. Pittsburgh, PA, 2002.
- CIGAS, J. Hands-on networking, rockhurt university. In: ACM SIGCOMM, WORKSHOP ON COMPUTER NETWORKING: CURRICULUM DESIGNS AND EDUCATIONAL CHALLENGES. Pittsburgh, PA, 2002.
- COMER, D. Networking curricula and laboratory, purdue university. In: ACM SIGCOMM, WORKSHOP ON COMPUTER NETWORKING: CURRICULUM DESIGNS AND EDUCATIONAL CHALLENGES. Pittsburgh, PA, 2002.
- GALLO, M. A.; HANCOCK, W. M. *Comunicação entre computadores e tecnologia de rede*. São Paulo: Pioneira Thomson Learning, 2003.
- KELEM, N.; FEIERTAG, R. *A Separation Model for Virtual Machine Monitors*. Ieee computer society symposium. Oakland, California - USA: Research in Security and Privacy, 1991. Proceedings., 1991.
- KUROSE, J. What should we teach in a first grad-level networking course?, university of massachusetts. In: ACM SIGCOMM, WORKSHOP ON COMPUTER NETWORKING: CURRICULUM DESIGNS AND EDUCATIONAL CHALLENGES. Pittsburgh, PA, 2002.
- KUROSE, J. et al. Workshop on computer networking: curriculum designs and educational challenges. *ACM SIGCOMM Computer Communications Review*, v. 32, n. 5, p. 1-9, November 2002.

NABHEN, R.; MAZIERO, C. Education for the 21st century - impact of ict and digital resources. In: _____. International federation for information processing. [S.l.]: Boston:Springer, 2006. v. 210, cap. Some Experiences in Using Virtual Machines for Teaching Computer Networks, p. 93–104.

POPEK, G. J.; GOLDBERG, R. P. Formal requirements for virtualizable third generation architectures. In: _____. International federation for information processing. New York, NY, USA: Communications of the ACM, 1974. v. 17, cap. Some Experiences in Using Virtual Machines for Teaching Computer Networks, p. 412 – 421.

ZARKI, M. Offering a hands-on computer networking course, irvine university. In: ACM SIGCOMM, WORKSHOP ON COMPUTER NETWORKING:CURRICULUM DESIGNS AND EDUCATIONAL CHALLENGES. Pittsburgh, PA, 2002.