

ALEXANDRE JORGE LOUREIRO FIGUERÊDO DE CARVALHO

LTSP – LINUX TERMINAL SERVER PROJECT

E A

INCLUSÃO DIGITAL

Monografia apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras, como parte das exigências do curso de Pós-Graduação *Lato Sensu* para obtenção do título de especialista em Administração de Redes Linux.

Orientadora

Profª. Kátia Cilene Amaral Uchôa

LAVRAS

MINAS GERAIS – BRASIL

2005

ALEXANDRE JORGE LOUREIRO FIGUERÊDO DE CARVALHO

LTSP – LINUX TERMINAL SERVER PROJECT

E A

INCLUSÃO DIGITAL

Monografia apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras, como parte das exigências do curso de Pós-Graduação *Lato Sensu* para obtenção do título de especialista em Administração de Redes Linux.

APROVADA em ____ de _____ de _____

Prof. _____

Prof. _____

Profª. Kátia Cilene Amaral Uchôa
(Orientadora)

LAVRAS
MINAS GERAIS – BRASIL

DEDICATÓRIA

Aos meus pais e minha esposa Jandelma, por todos os momentos de incentivo e desespero que passaram durante o período de laboratório deste trabalho. Ao meu amigo João Almeida e Silva, que participou também de vários testes para este projeto. A todos os professores e colegas do curso de Pós-Graduação em Administração de Redes Linux, da Universidade Federal de Lavras/MG – UFLA, por todo este afeto e respeito pelo mundo livre.

AGRADECIMENTOS

Este projeto deve sua existência primeiramente a Deus, a diversos amigos que administram (ou estão ligados a) pequenos negócios e também alguns alunos meus do curso técnico de informática, especialmente o curso de Linux. Certamente, sempre lhes serei grato e espero que esta monografia os ajude de alguma maneira nas suas vidas.

Gostaria de agradecer especialmente, o João Almeida e Silva, também aluno da turma ARL2003S1, que foi parceiro neste trabalho e que também realizou a sua monografia, com certeza, uma pessoa que eu precisava conhecer para abrir os horizontes no mundo do *Software Livre*.

Paulo, da Catende Link, que forneceu Internet completamente de graça para os testes no laboratório dentro da minha residência.

À Tec Center Informática, que possibilitou a minha pessoa ministrar os cursos de Linux I, II e III. Em especial ao seu Diretor, Sr. Marcos Roque. À CETAP Informática onde sou Diretor e Professor.

Aos meus familiares, como sempre, por darem-me apoio nos momentos mais importantes da minha vida. Minha esposa, Jandelma, que foi extremamente compreensiva em todos os momentos de desespero no período de testes. Mariane, minha filhinha de três anos, que já começou a brincar com os programinhas de Linux.

RESUMO

O presente projeto tem o objetivo de levar a tecnologia da informação ao cidadão de forma simples e irrestrita através de telecentros, escolas informatizadas ou terminais públicos de informática a fim de possibilitar a interação com os mais diversos segmentos da sociedade, a fim de usufruir dos vários serviços tecnológicos, tais como: conta de *e-mail*, *e-commerce*, ensino à distância, etc.

Além do mais, o material tem o intuito de servir como modelo para instituições, tanto públicas como privadas, ou ainda, para pessoas que estejam interessadas em fazer algo pelo social. Busca resumir o que se precisa para ter um centro de TI de alta qualidade com gastos reduzidos de *hardware* e gasto zero em *software*. É consideravelmente inviável fazer a inclusão digital com “*Software Proprietário*” tendo em vista o grande gasto com pagamento de licenças em programas. Com o “*Software Livre*” pode-se gerar independência tecnológica no país a um baixo custo e gerando riquezas para a própria comunidade local.

A intenção do projeto é fornecer um material partindo do princípio de uso com *Software Livre* adotando o sistema operacional GNU/Linux, ferramenta *Office* para escritório, alguns programas extras e, principalmente, a ferramenta *LTSP – Linux Terminal Server Project*, o qual possibilitará a integração de máquinas antigas para trabalhar como *desktop* ligadas a um servidor principal.

SUMÁRIO

Capítulo I - Introdução	1
Capítulo II – A Questão Social	4
2.1 – O que é a Inclusão Digital	4
2.2 – A importância da Inclusão Digital	6
2.3 – Inclusão Digital no Brasil	7
2.4 – Inclusão Digital com <i>Software Livre</i>	9
2.5 – Inclusão Digital com GNU/Linux	12
2.6 – Inclusão Digital e o Governo	13
Capítulo III - A Parte Física	15
3.1 – Placas de rede	15
3.2 – Cabos da rede	16
3.3 – Hubs e Switchs	18
3.4 – O Servidor	19
3.5 – As Estações de Trabalho	21
3.6 – Adoção do GNU/Linux como Sistema Operacional	22
Capítulo IV – Montagem do Servidor e Estações de Trabalho	24
4.1 – <i>LTSP – Linux Terminal Server Project</i>	24
4.2 – Entendendo o <i>LTSP</i>	25
4.3 – Instalando os Serviços	27
4.4 – Pacotes do <i>LTSP</i>	30
4.5 – Configuração dos Terminais	35
4.6 – Configuração do Servidor	37
4.7 – Alguns Problemas	49
Capítulo V – Experiência em Sala de Aula.....	51
5.1 – A Sala de Aula	51
5.2 – O Início do Projeto LTSP e Inclusão Digital no CETAP	51
5.3 – Introdução ao Processamento de Dados	52
5.4 – <i>Debian GNU/Linux Sarge 3.1</i>	52
5.5 – <i>OpenOffice</i>	53
5.6 – Navegador Internet <i>Mozilla</i>	55
5.7 – Tarefa Realizada	55
Conclusões	56
Referências Bibliográficas	58
Anexo A (Arquivos de Configuração)	59

LISTA DE FIGURAS

Figura 3.1: Placa de Rede <i>Ethernet</i> 10/100 Megabits	15
Figura 3.2: Cabo UTP 5e	16
Figura 3.3: Padrão EIA 568B	17
Figura 3.4: <i>Switch</i> de 16 portas	18
Figura 3.5: O Servidor <i>LTSP</i>	29
Figura 3.6: Algumas Estações de Trabalho	21
Figura 4.1: Página Oficial do Projeto <i>LTSP</i>	24
Figura 4.2: Aplicativo <i>gdmsetup</i>	30
Figura 4.3: Aplicativo <i>ltsadmin</i>	32
Figura 4.4: <i>ltsadmin</i> Escolhendo os Aplicativos	33
Figura 4.5: Diretório <i>/opt/ltsp/i386</i>	33
Figura 4.6: Configurações no <i>ltsadmin</i>	34
Figura 4.7: Página Oficial do Projeto <i>Rom-o-Matic</i>	36
Figura 4.8: Processo de <i>Boot</i> no Terminal	37
Figura 4.9: Arquivo <i>/etc/dhcp3/dhcpd.conf</i> – Parte 1	38
Figura 4.10: Arquivo <i>/etc/dhcp3/dhcpd.conf</i> – Parte 2	39
Figura 4.11: Arquivo <i>/etc/dhcp3/dhcpd.conf</i> – Parte 3	40
Figura 4.12 Arquivo <i>/etc/exports</i>	41
Figura 4.13: Arquivo <i>/etc/hosts.allow</i>	42
Figura 4.14: Arquivo <i>/opt/ltsp/i386/etc/lts.conf</i> – Parte 1	43
Figura 4.15: Arquivo <i>/opt/ltsp/i386/etc/lts.conf</i> – Parte 2	44
Figura 4.16: Arquivo <i>/etc/hosts</i>	48
Figura 4.17: Tela de <i>Login</i> do <i>GDM</i>	49
Figura 4.18: Terminal sem <i>HD</i>	50
Figura 5.1: Terminal rodando o <i>OpenOffice Writer</i>	51

CAPÍTULO I

INTRODUÇÃO

Este trabalho tem como objetivo mostrar, através de estudo de casos, que o *Software Livre* é uma fonte de vantagem estratégica para as empresas, trazendo benefícios práticos, independentemente de posições filosóficas ou princípios morais. O texto abrange os principais aspectos de uso do *LTSP (Linux Terminal Server Project)* na criação de um pequeno escritório / sala de aula produtiva, com finalidades de uso em projetos de inclusão digital. Pretende-se abordar vários aspectos da montagem de rede e configuração do servidor GNU/Linux (especialmente a distribuição *Debian*) como servidor para terminais leves.

O projeto foi efetivado como parte integrante do curso técnico de informática na empresa CETAP – Consultoria e Informática, localizada na cidade de Catende/PE. A idéia partiu do ponto de vista de que necessitava de instruir jovens que não têm condições em pagar cursos de computação. Para tal tarefa foi adquirido um computador *Athlon* para funcionar como servidor e 6 máquinas bem menores, para funcionarem como terminais leves.

A adoção do GNU/Linux para este caso é clara, a melhor forma de incentivar a inclusão digital é com o uso de *Software Livre*. Além do mais, foi a única solução que se achou para utilizar máquinas realmente antigas para servirem como terminais.

A monografia está dividida em cinco capítulos principais. As seções são extremamente objetivas, passando dos componentes básicos de uma estrutura

rede para os pontos mais importantes de como trabalhar com o GNU/Linux de maneira fácil e eficiente. Veja a seguir o que pode ser encontrado em cada capítulo:

O Capítulo II retrata **A Questão Social**. De certa forma é uma parte bem teórica. Aborda alguns aspectos da sociedade atual, inclusive sobre a educação digital em nosso país, a evolução do povo em relação à atualidade, direitos do cidadão e outros. Além disso, discorre a respeito dos vários projetos de inclusão digital para a sociedade. Por fim, este capítulo falará do futuro promissor que o País terá com a inclusão digital sendo posta para todas as pessoas da nação.

Já no Capítulo III - **A Parte Física** mostrará o material de como montar uma rede de computadores. Este capítulo fornece algumas dicas de como interligar computadores, tratando de adaptadores de rede, *hub*, *switch*, cabos, conectores e alguns aspectos de Internet. Também será ilustrado o *hardware* do servidor e das estações de trabalho. Por fim, será mostrado que o sistema operacional GNU/Linux é uma saída viável e com alta qualidade em relação aos sistemas operacionais de caráter proprietário.

O Capítulo IV – **Montagem do Servidor e Estações de Trabalho**. Será mostrada a configuração e os aspectos de um servidor para terminais leves, entrando no assunto *LTSP*, e também, mostrará como configurar as estações de trabalho para se conectar ao servidor principal.

Capítulo V – **Experiência em Sala de Aula**. Neste ponto, o projeto tem como principal foco a prática do aluno na informática. Aqui será mostrado os resultados da sala de aula usando terminais leves. O aluno tendo o seu

primeiro contato com a informática e com alguns aplicativos, introduzindo-o ao Processamento de Dados o que propicia noções mais acuradas de *hardware* e *software*. Depois, os conceitos básicos de sistema operacional e como prática o *Debian GNU/Linux Sarge 3.1*. Após isso, o pupilo verá alguns aplicativos adicionais, principalmente os voltados para o escritório como editor de texto, planilha de cálculo e acesso à Internet.

CAPÍTULO II

A QUESTÃO SOCIAL

2.1 O Que é a Inclusão Digital

A inclusão digital e o combate à exclusão social e econômica estão intimamente ligados, em uma sociedade onde cada vez mais o conhecimento é considerado riqueza e poder. Se há um consenso a respeito do que é inclusão digital é que o desenvolvimento socio-econômico e político deste início de século XXI passa também pelo domínio das chamadas TICs — tecnologias de informação e comunicação, então qualquer pessoa necessita ter conhecimento dessas novas tecnologias. Pode-se observar que sociedade rica usa com intensidade as redes informacionais para se comunicar e para armazenar e processar informações, enquanto os países pobres e em desenvolvimento têm suas populações distantes dos benefícios das redes informacionais.

Como escreveu [Castells (2003)], em seu livro sobre a grande importância da Internet na sociedade:

“Desenvolvimento sem a Internet seria o equivalente à industrialização sem eletricidade na era industrial. É por isso, que a declaração freqüentemente ouvida sobre a necessidade de se começar com “os problemas reais do Terceiro Mundo” — designando com isso: saúde, educação, água, eletricidade e assim por diante — antes de chegar à Internet, revela uma profunda incompreensão das questões atuais relativas ao desenvolvimento. Porque, sem uma economia e um sistema de administração baseados na Internet, qualquer país tem pouca chance de gerar os recursos necessários para cobrir suas necessidades de desenvolvimento, num terreno sustentável — sustentável em termos econômicos, sociais e ambientais”.

A inclusão digital deve favorecer a apropriação da tecnologia de forma consciente, que torne o indivíduo capaz de decidir quando, como e para que utilizá-la. Do ponto de vista lógico, a inclusão digital significa aplicar as tecnologias a processos que contribuam para o fortalecimento de suas atividades econômicas, de sua capacidade de organização, do nível educacional e da auto-estima de seus integrantes, de sua comunicação com outros grupos, de suas entidades e serviços locais e de sua qualidade de vida [Cruz (2005)].

Iniciativas de promoção da inclusão estariam, então, diretamente relacionadas à motivação e à capacidade para a utilização das TICs de forma crítica e empreendedora, objetivando o desenvolvimento pessoal e comunitário. A idéia é que, apropriando-se destes novos conhecimentos e ferramentas, os indivíduos possam desenvolver uma consciência histórica, política e ética, associada a uma ação cidadã e de transformação social, ao mesmo tempo em que se qualificam profissionalmente. É neste contexto que se propõe discutir a inclusão digital e as ações de responsabilidade social das empresas e instituições.

Doar computadores, periféricos e recursos financeiros, prover a conectividade e encorajar o voluntariado interno são apenas algumas formas de promover a inclusão digital como ação de responsabilidade social. Incentivar a produção e a troca de conhecimento nas comunidades localizadas na área de entorno da empresa; fornecer dicas profissionais, compartilhar experiências, elaborar projetos em conjunto; incentivar e influenciar a busca de auto-sustentabilidade das comunidades; incentivar o empreendedorismo e fornecer apoio tecnológico também são, hoje, valiosas ações corporativas que contribuem para a prática da responsabilidade social, favorecendo a inclusão digital e, conseqüentemente, a social [Cruz (2005)].

Do ponto de vista de uma empresa ou instituição com responsabilidade social, investir em programas de inclusão digital significa entender “solidariedade” não só como mero conceito assistencialista, mas como promoção de oportunidades para a produção e a disseminação de conhecimento e renda.

2.2 A Importância da Inclusão Digital

O acesso às tecnologias da informação e da comunicação, também chamado inclusão digital, está diretamente relacionado, no mundo atual, aos direitos básicos à informação e à liberdade de opinião e expressão. A exclusão digital é uma das muitas formas de manifestação da exclusão social. Não é um fenômeno isolado ou que possa ser compreendido separadamente, pois trata-se de mais uma conseqüência das diferenças já existentes na distribuição de poder e de renda. Num momento em que empresas e governos migram informações e serviços para os meios eletrônicos, o excluído digital passa a ter dificuldade de conhecer e de exercer seus direitos de cidadão.

A inclusão digital não se resume à disponibilidade de computadores e de telefones, mas à capacitação das pessoas para o uso efetivo desses recursos tecnológicos. Conforme destacado por [Cruz (2005)], para ser incluído digitalmente, não basta ter acesso a micros conectados à Internet. Também é preciso estar preparado para usar estas máquinas, não somente com capacitação em informática, mas com uma preparação educacional que permita usufruir de seus recursos de maneira plena.

“As tecnologias da informação e da comunicação precisam se tornar ferramentas que contribuam para o

desenvolvimento social, intelectual, econômico e político do cidadão”.

2.3 Inclusão Digital no Brasil

A massificação do uso das Tecnologias de Informação e de Comunicação para modernização do setor produtivo e das atividades estatais criou uma nova classe de excluídos socialmente: **os digitalmente excluídos**. Trata-se de cidadãos que necessitam de emprego, mas que não têm *e-mail* para fornecer numa entrevista; que precisam de uma certidão negativa da administração pública municipal, mas não fazem idéia de que pode ser obtida fora do ambiente físico em que esta se materializa; e que nunca realizaram uma ligação telefônica:

“Muitos dirigentes públicos e empresariais ainda acham que o uso do computador só é importante para a profissionalização da população. Essa visão constitui a cultura do uso limitado da tecnologia e deixa de lado a dimensão da cidadania. Sem dúvida, é urgente e prioritário implantar laboratórios de informática em todas as escolas e conectá-las à rede informacional”[Silveira (2003)].

Entende-se que o acesso a recursos tecnológicos para educação, trabalho, lazer e exercício da cidadania encerra, genericamente, o significado do termo “**Inclusão Digital**”, colocando-nos perante os olhos do desafio estratégico para o desenvolvimento nacional. Atualmente, pensar em alternativas para a solução de problemas sociais inteiramente dissociadas do uso de TICs implica negligenciar importante ferramenta, a qual pode estender sobremaneira o raio de atuação de uma política social ou de iniciativa comunitária para diminuição das desigualdades sociais. O trabalho realizado por inúmeras instituições de educação e capacitação (CDI, Prefeitura de São Paulo,

GESAC), no Brasil, atesta este entendimento.

A implementação deste ideal passa, inicialmente, pelo acesso a computadores pessoais e de uso coletivo. No Brasil, apenas 5% da população tem acesso à Internet, e esse número representa um dos maiores índices entre os países em desenvolvimento. Em países desenvolvidos como EUA, Canadá e Inglaterra, esse percentual fica entre 30 e 50% da população e se encontra em crescente expansão. Segundo o Livro Verde da Sociedade da Informação no Brasil, o grande desafio de países emergentes "é a aplicação das tecnologias da informação e comunicação nos níveis médio, graduação e pós-graduação" [Takahashi (2005)].

A capacitação em computador consiste na chamada alfabetização digital, que objetiva fornecer aos cidadãos uma proficiência para utilização dos recursos tecnológicos disponíveis, como: a operação de computadores e a utilização de programas e aplicativos. Este é o primeiro passo para que os cidadãos possam acessar e utilizar estas ferramentas com uma finalidade própria, no sentido de estimular o aprendizado e, conseqüentemente, o exercício da cidadania. Ela se baseia na familiarização das pessoas com o ciberespaço e no aprendizado para a localização das informações desejadas.

Pode-se adaptar à realidade brasileira, a frase de Pierre Levy que diz:

"Ainda que as pessoas aprendam em suas experiências profissionais e sociais, ainda que a escola e a universidade estejam perdendo progressivamente seu monopólio de criação e transmissão do conhecimento, os sistemas de ensino público podem ao menos dar-se por nova missão a de orientar os percursos individuais no saber e contribuir para o reconhecimento do conjunto de know-how das pessoas, inclusive saberes não-acadêmicos" [Levy (1999)].

Não há necessidade de se estar vinculado a instituições de ensino para ter direito às prerrogativas da tecnologia. Não há necessidade de ser *experts* em computação para exercer o direito de usuário. E ainda, não há necessidade de ser um profissional para utilizar a rede mundial. Todos têm direito de usar a Internet.

A criação de ambientes virtuais de conhecimento, por sua vez, baseia-se em discussões recentes sobre usabilidade e ergonomia de portais, associadas a estudos avançados em organização e recuperação do conhecimento, nos quais o Brasil tem ocupado lugar de destaque desde a última década. Essa criação tem como escopo tentar solucionar um dos maiores desafios da Sociedade da Informação: o gerenciamento de uma grande variedade de informações, apoiada em livros, bases de dados, *e-mails*, relatórios, notícias, estudos acadêmicos, etc. A disponibilidade de informações na *web* possibilitou à população ter acesso a boas informações e fontes sobre qualquer assunto. Mas também colocou um obstáculo àqueles que não têm paciência, ou não são especialistas em gerenciamento e recuperação de informações. Pode-se dizer que, a dificuldade em encontrar respostas satisfatórias às dúvidas e problemas cotidianos é um fato cada vez mais presente na sociedade da informação [Cruz (2005)].

2.4 Inclusão Digital com *Software Livre*

A inclusão digital sendo realizada com *software* (programas de computadores) de cunho proprietário terá um efeito econômico bastante alto. Esta filosofia retorna à sistemática da utilização de programas completamente presos por suas formas e conceitos, inclusive pelos leoninos contratos de

licenças.

Com a difusão da Internet, o movimento de *Software Livre* ganhou o mundo e logrou produzir um breve sistema operacional livre, completo e multifuncional, o GNU/Linux. A história da origem do GNU/Linux tem-se que, o finlandês Linus Torvald conseguiu compilar todos os programas e ferramentas do movimento *GNU*¹ em um *kernel*, um núcleo central, o que viabilizou o sistema operacional.

No mundo livre as coisas são bem mais interessantes, produtivas e econômicas, sem falar que alternativas de adaptações ou customizações ficam a critério de cada usuário. Isto relembra “*A Catedral e o Bazar*”, [Raymond (1999)], em que o autor compara dois estilos completamente distintos de desenvolvimento de *software*, o modelo comercial denominado catedral e o modelo de código aberto denominado bazar. Neste último, qualquer um com acesso à Internet e habilidades de programação pode integrar ao processo de desenvolvimento do *software*. No modelo bazar, as versões betas são lançadas e testadas diariamente pela comunidade distribuída na rede. Os *bugs* são rapidamente descobertos e o *software* é constantemente melhorado.

A cada ano que passa, o *Software Livre* ganha mais força, movido pela paixão de milhares de desenvolvedores e milhões de usuários que acreditam em uma ética segundo a qual o conhecimento não deve permanecer oculto, mas ser compartilhado. Nesta ótica, este trabalho tem como objetivo mostrar, através de estudo de casos, que o *Software Livre* é uma fonte de vantagem estratégica para as empresas, trazendo benefícios práticos, independentemente de posições

1 - *GNU (GNU's Not Unix)* movimento criado por Richard Stallman em 1985.

filosóficas ou princípios morais.

A liberdade para usar, copiar, modificar e redistribuir *Software Livre* confere-lhe uma série enorme de vantagens sobre o *Software Proprietário*. A comunidade de desenvolvimento de *Software Livre* está espalhada pelo mundo todo e seus participantes cooperam nos projetos através da Internet. Estima-se que participam desta comunidade mais de 100 mil programadores e projetistas, com a grande maioria deles trabalhando voluntariamente em um ou mais projetos. Estima-se também que existam mais de 12 milhões de usuários regulares de sistemas operacionais e aplicativos distribuídos como *Software Livre*. Recentemente, empresas como *IBM* e *Hewlet-Packard* passaram a investir no desenvolvimento de *software* a ser distribuído livremente, bem como em serviços para usuários de *Software Livre*. [Moore (2000)].

Com números desta ordem de grandeza, e produtos de excelente qualidade, o modo de produção de *Software Livre* aparenta ser um novo e fundamental componente da economia moderna.

Em suma, é fundamental integrar a política de inclusão digital com o uso de *Software Livre*, pois, desta forma, estará gerando uma grande economia em razão do não pagamento de licenças, e ainda, reflete ao usuário o estímulo da pesquisa e do aprendizado, uma vez que o código fonte está aberto possibilitando a todos por interesses do conhecimento.

2.5 Inclusão Digital com GNU/Linux

Quando se considera a dificuldade de obtenção de *software* confiável no modo tradicional, e que os sistemas empregados são cada vez mais complexos, sofisticados e imprescindíveis, a existência de um modo alternativo de produção de sistemas, e de qualidade freqüentemente superior àquela de modo tradicional, adquire importância estratégica para o País. A difusão do uso de *Software Livre* no Brasil traria conseqüências importantes, destacando-se dentre elas um incremento qualitativo na capacidade nacional de produzir *software* sofisticado e de boa qualidade.

A qualidade técnica do sistema operacional GNU/Linux advém do modo de produção do *Software Livre*, que envolve grandes números de desenvolvedores espalhados pelo mundo. Por sua vez, esta mesma qualidade técnica atrai novos usuários, vários dos quais passam a agir como testadores e desenvolvedores do sistema. Esta atuação produz melhorias na qualidade do sistema, o que acaba por atrair novos usuários. Este ciclo virtuoso deve-se à existência de vários programadores e *hackers*, em nível mundial, tanto de usuários como de desenvolvedores envolvidos com diferentes projetos de programas livres..

Já os benefícios econômicos são muito maiores e mais importantes do que o simples pagamento de licenças de *software*. A robustez e confiabilidade do *Software Livre* provocam reduções significativas em custos operacionais. A disponibilidade do código fonte permite que os sistemas sejam adaptados às condições e necessidades dos usuários. Estas adaptações poderão ser efetuadas por profissionais brasileiros, que terão oportunidades de desenvolvimento muito distintas daquelas vigentes num mercado mono-político. Além disso, a

possibilidade de consulta ao código dos programas permite condições de estudo e aprendizado que são absolutamente inviáveis com *Software Proprietário*.

O código de um programa distribuído como *Software Livre* torna-se um bem público que está à disposição de toda a sociedade. Neste sentido, *software* assemelha-se ao conhecimento científico, que uma vez difundido pode ser livremente utilizado por todos, e que assim possibilita o próprio avanço da Ciência. Portanto, os benefícios sociais da publicação e do uso de *Software Livre* são a liberdade na utilização das ferramentas, e especialmente na disponibilidade do conhecimento envolvido na produção destas ferramentas, bem como de sua evolução.

Outro benefício social importante é a transparência na codificação das informações tratadas pelos programas. Os formatos empregados para armazenar e tratar as informações são abertos porque o código fonte dos programas pode ser livremente examinado. Com isso, inexistente a possibilidade de que, por exemplo, dados usados no serviço público sejam mantidos em formatos de propriedade de uma entidade privada. O mesmo raciocínio se aplica aos protocolos de comunicação empregados para a transferência de informações entre computadores ou sistemas.

2.6 Inclusão Digital e o Governo

Em vista do que foi exposto até aqui, os três níveis de governo têm um papel fundamental na criação das condições necessárias ao estabelecimento de usuários e desenvolvedores de *Software Livre* no Brasil. Os governos devem incentivar a utilização deste modelo em suas repartições e autarquias, desta

forma propiciando o estabelecimento de mais e mais usuários. Estes por sua vez provocarão a demanda por treinamento e manutenção dos sistemas em uso. O poder de compra dos três níveis de governo, e em especial o do Governo Federal, deve ser empregado para criar a demanda pelo desenvolvimento de aplicativos que atendam às suas necessidades nas áreas de gestão, tributação, educação e governo-eletrônico, para citar apenas áreas de maior visibilidade.

Uma outra estratégia do governo, e com forte teor de impacto foi a criação do programa de inclusão digital “Computador para Todos” [Medida Provisória nº 252, de 15 de junho 2005]. Nesse programa o governo faz parcerias com empresas e com o setor de comunicação para disponibilizar máquinas com preços reduzidos e linhas de crédito ampliadas. As referidas máquinas são equipadas com *Software Livre* e ainda têm benefícios nos custos para a navegação na Internet.

“Micros de até R\$ 1.400 que seguem as configurações estipuladas pelo governo poderão ser parcelados em até 24 prestações de R\$ 70. O valor final a ser pago inclui os juros cobrados na venda a prazo.

Para se encaixar nessa categoria, o equipamento, de qualquer marca, deve utilizar obrigatoriamente Software Livre, como o GNU/Linux. Além disso, precisa contar com um processador de 1,5 GHz, disco rígido de 40 GB, memória RAM de 128 MB, monitor de 15 polegadas, unidade de disco flexível, unidade de CD-ROM, modem de 56 K, placa de vídeo, áudio e rede on-board, mouse, teclado e porta USB e 26 programas” [Carpanez (2005)].

Finalizando, está mais que na hora do Brasil alavancar no modelo de *Software Livre* em benefício dos seus cidadãos.

CAPÍTULO III

A PARTE FÍSICA

Os componentes básicos para montar a rede no projeto foram: placas de rede 10/100, cabos UTP padrão 5e com conectores RJ45 e um *switch de 16 portas*. É uma rede bem simples de classe “C” que qualquer pessoa com conhecimento razoável em informática consegue montar.

3.1 Placas de Rede



Figura 3.1: Placa de Rede *Ethernet 10/100 Megabits*

O padrão utilizado neste projeto (e é o padrão mundial) foi o *Ethernet* (Figura 3.1) para as placas de redes. É bom frisar neste ponto, que *Ethernet* é visto por muita gente como se fosse uma placa de rede, pela sua enorme utilização. *Ethernet* é um nome de um padrão que diz como os dados são transmitidos. Todas as placas que seguem este padrão são chamadas de placas *Ethernet*. Quando se compra uma placa, por exemplo, uma *realtek8139* encore. Neste caso, tem-se aí o modelo/marca da placa, e o padrão é o *Ethernet*.

Para a montagem da rede foi utilizado, mais precisamente, placas de rede *Ethernet* de 100 *Megabits* (as famosas 10/100, por possibilitar também usar redes de 10 *Megabits*). Neste ponto, é de grande importância frisar que, se

existir na rede apenas um micro com uma placa de 10 *Megabits*, este trabalhará nesta velocidade. É bom ter o devido cuidado com as mencionadas placas, pois o ideal é placa de 100 *Megabits*.

As redes de 100 *Megabits* são padrões hoje em dia, porém já existem placas de até 10 *Gigabits*. O padrão atual possibilita ter placas a um custo bem baixo (placas de rede 10/100 são vendidas por menos de 20 reais, e muitas placas-mãe já as trazem inclusas também).

3.2 Cabos da Rede

Existem basicamente três tipos diferentes de cabos de rede: os cabos de par trançado (os mais utilizados), os cabos de fibra óptica (para longa distância) e os cabos coaxiais, usados em redes antigas. Utilizou-se aqui cabos de par trançado categoria 5e. Existem cabos de categorias 1 até categoria 7. Como os cabos de categoria 5 são suficientes para as redes de 100 *Megabits*, eles são os mais comuns e também os mais baratos; custam menos de 1 real o metro. Os cabos categoria 5e seguem um padrão um pouco mais estrito, eles são a preferência.



Figura 3.2: Cabo UTP 5e

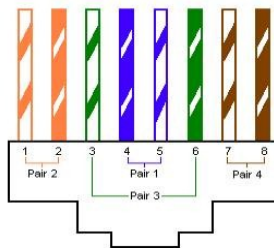
É bom ressaltar que a distância máxima permitida é de 100 metros. O cabo de par trançado cat5e é o tipo de cabo usado atualmente, com qualidade

superior aos cat5. Eles oferecem uma taxa de atenuação de sinal mais baixa, o que ajuda na distância, próximo dos 100 metros admitidos. Caso haja dúvida em relação à categoria, basta checar as inscrições decalcadas no cabo, entre elas está a categoria do cabo, como mostra a figura 3.2.

O que será crimpado nas extremidades do cabo chama-se “Conector RJ-45”. Para esta atividade deve-se ter o alicate específico (alicate de crimpagem). Este alicate esmaga os contatos do conector, fazendo com que eles entrem em contato com os fios do cabo de rede.

A ordem correta dos fios (padrão EIA 568B), conforme mostra a figura 3.3 é a seguinte:

- 1 – Branco com Laranja
- 2 – Laranja
- 3 – Branco com Verde
- 4 – Azul
- 5 – Branco com Azul
- 6 – Verde
- 7 – Branco com Marrom
- 8 – Marrom



Standard EIA/TIA T568B Wiring Diagram

Figura 3.3: Padrão EIA 568B

Os citados cabos são encaixados nesta ordem, com a trava do conector virada para baixo. Na hora de crimpar é preciso fazer um pouco de força para que o conector fique firme. A função do alicate é fornecer pressão suficiente para que os pinos do Conector RJ-45, que internamente possui a forma de lâminas, esmaguem os fios do cabo, alcançando o fio de cobre e criando o contato.

3.3 Hubs e Switchs

Servem como um ponto central, permitindo que todos os pontos se comuniquem entre si. As placas de rede são ligadas ao *hub* ou *switch* (figura 3.4), e ainda, é possível ligar vários *hubs* ou *switchs* entre si, para aumentar o número de máquinas numa rede.



Figura 3.4: *Switch* de 16 portas

A grande diferença entre *hubs* e *switchs* é que o *hub* apenas retransmite tudo o que recebe para todos os micros conectados a ele, como se fosse um espelho. Isso significa que apenas um micro pode transmitir dados de cada vez e que todas as placas precisam operar na mesma velocidade, que será sempre nivelada por baixo.

Já os *switchs* são aparelhos muito mais inteligentes. Eles fecham canais exclusivos de comunicação entre o micro que está enviando dados e o que está recebendo, permitindo que vários pares de micros troquem dados entre si ao mesmo tempo. Outra grande vantagem dos *switchs* é que em redes onde são misturadas placas 10/10 e 10/100, as comunicações podem ser feitas nas velocidades das placas envolvidas.

Atualmente, os *hubs* são *hub-switchs*, modelos de *switchs* mais baratos. Depois destes, tem-se os *switchs* de verdade, capazes de gerenciar um número

muito maior de portas, sendo adequados para redes maiores. No projeto foi utilizado um *hub-switch* 10/100 de 16 portas.

3.4 O Servidor



Figura 3.5: O Servidor *LTSP*

O servidor *LTSP* (figura 3.5) deve ter um *hardware* bem mais robusto do que as estações de trabalho (estas, por sua vez, podem ser máquinas pequenas e simples, como um *Pentium Standard*, ou até mesmo um 80486 com 16 MB de memória). O servidor utilizado para os testes do projeto tem a seguinte configuração:

- *AMD Athlon 1.8 GHz*
- *HD 40 GB*
- *512 de Ram*
- *Drives de CD-RW e Disquete*
- *Monitor de 17"*

A configuração do servidor vai depender muito do tamanho da rede e

dos aplicativos utilizados. Obviamente, um servidor com 1 *GB* de memória *RAM*, dividido entre 20 terminais, executa em geral os aplicativos com um desempenho muito melhor que um *desktop* com 256 *MB* usado por uma única pessoa. A configuração mínima para atender a 10 terminais seria um *Pentium III* ou *Athlon* com 512 *MB* de *RAM*. O ideal é monitorar a utilização do processador e da memória *RAM* livre durante um tempo. Conforme for necessário, pode-se adicionar memória ou colocar um processador um pouco maior.

A memória *RAM* é compartilhada de uma maneira bastante interessante. Os aplicativos são carregados na memória do servidor apenas uma vez, independentemente do número de usuários que o utilizarem simultaneamente. O sistema carrega o aplicativo uma vez, e depois passa a abrir diferentes seções do mesmo programa, o que faz com que o carregamento seja bem mais rápido e o uso da memória seja otimizado.

Uma outra característica importante é o desempenho e a capacidade do *HD*. Tudo será armazenado no servidor, por isso é importante que o *HD* seja rápido e tenha bastante espaço livre.

O sistema operacional utilizado no projeto foi o *Debian GNU/Linux Sarge 3.1*. *Debian* é uma distribuição bem completa e sendo também os seus aplicativos bastante testados. A instalação do sistema operacional segue o padrão normal (aqui não será abordado o processo de instalação do GNU/Linux). O ideal é que se tenha uma boa distribuição instalada e ainda um *firewall* como o *iptables* em funcionamento, tendo em vista algumas vulnerabilidades nos serviços do servidor, caso a máquina servidora seja

também um *gateway* para acesso à Internet.

3.5 As Estações de Trabalho

Para as estações de trabalho foram utilizados equipamentos bem superados para os aplicativos atuais (este ponto demonstra a grande economia que se tem com a utilização de terminais leves com GNU/Linux). São máquinas entre *Pentium 100 e K6 II 500*, utilizando-se 64 MB de memória em geral.



Figura 3.6: Algumas Estações de Trabalho

O fato dos terminais serem de pequena potência reflete necessariamente o impacto de processamento no servidor (Figura 3.6). O servidor fica com todo o trabalho, que é executar os programas e armazenar todos os dados. Ele envia para os clientes apenas instruções para montar as janelas que serão exibidas, e estes enviam de volta os movimentos do *mouse* e as teclas digitadas no teclado.

A configuração mínima para os terminais é um 486 com 8 MB, e a

configuração ideal é um *Pentium* 100 com 16 MB. Em teoria, pode-se utilizar até mesmo um 386 como terminal, mas neste caso a demora na atualização das janelas será enorme. Caso o processador seja muito lento, a estação tem dificuldades para fazer a atualização de tela e as respostas começam a ficarem muito lentas.

O *boot* da estação é bem rápido, demora menos de 30 segundos (no *Pentium*) para cair na tela de *login* do servidor e, a partir daí, o tempo de carregamento do *KDE*, *Gnome* ou qualquer outro gerenciador de ambiente depende apenas do desempenho do servidor. A parte mais interessante é que se pode continuar usando micros antigos (os que já dispõem) e mesmo assim obter um bom desempenho, investindo apenas num servidor mais rápido.

3.6 Adoção do GNU/Linux Como Sistema Operacional

A adoção do GNU/Linux num projeto como este não é em vão. Trata-se de uma estratégia econômica e principalmente em nível de estudo/aprendizagem. Pessoas com grau avançado de informática podem até mesmo montar o seu próprio GNU/Linux, ou até mesmo produzir a sua própria distribuição.

Seria meio esquisito montar um projeto de inclusão digital com *software* de caráter puramente comercial. Ou seja, o investimento seria enorme num projeto deste tipo executando o *Terminal Server* da *Microsoft*.

O *LTSP* – *Linux Terminal Server Project* reúne tudo o que se precisa

para fazer máquinas com pouco poder de processamento trabalhar da mesma forma que computadores atuais. E o desprendimento para esta melhoria reduziria-se apenas na busca pelo aprendizado do funcionamento da tecnologia *LTSP*.

Muitos falam que o GNU/Linux ainda não está pronto para o *Desktop*. O autor do projeto discorda plenamente desta corrente e ainda questiona: O que é um *desktop*? Se um *desktop* é uma máquina rodando um sistema operacional prático, com editores de texto, planilha de cálculo, navegadores, programas de imagens, músicas, vídeos e ainda ferramentas de programação, o GNU/Linux não só está pronto para o *desktop* como é uma solução bastante econômica e estável.

CAPÍTULO IV

MONTAGEM DO SERVIDOR E ESTAÇÕES DE TRABALHO

4.1 LTSP – Linux Terminal Server Project

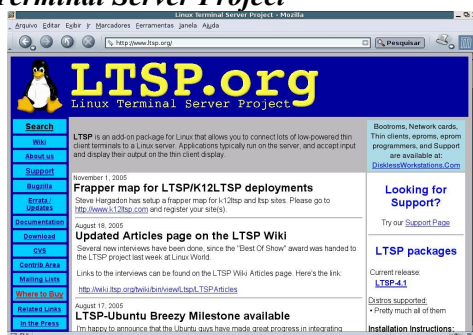


Figura 4.1: Página Oficial do Projeto LTSP

O *LTSP – Linux Terminal Server Project*², é uma solução bastante utilizada para a criação de terminais leves com o GNU/Linux. Ele utiliza uma combinação de serviços: *DHCP*, *TFTP*, *NFS* e *XDMCP*, para permitir que as estações executem aplicativos instalados no servidor. Os terminais dão *boot* (inicializa) via rede, baixando do servidor todos os programas que forem necessários.

O mais interessante de tudo é que não precisa ter *HD* (disco rígido) instalado nos terminais, apenas um *drive* de disquete ou CD e placa de rede para poder inicializar o equipamento. O *boot* é dado com a ajuda do *Etherboot*, um *software* minúsculo que permite que as estações dêem *boot* através da rede, obtendo todos os programas a partir do servidor. O *software* tem entre 35 e 40 *KB*, dependendo do *driver*, e é lido apenas uma vez, no início do *boot*.

As estações ainda podem ser iniciadas com um *chip* de *boot* colocado

2 - <http://www.ltsp.org> (Página Oficial do Projeto LTSP)

na placa de rede ou ainda via *PXE* (recurso utilizado em muitas placas *onboard* atuais), mas estes itens não serão tratados nesta seção.

Carlos Eduardo Morimoto faz uma feliz análise a respeito de terminais:

“Esta é uma possibilidade interessante. Os PCs evoluíram rápido nas últimas décadas, gerando pilhas de equipamentos obsoletos, que ainda funcionam mas são lentos demais para rodar os aplicativos atuais.

Ao invés de ter o trabalho de instalar o sistema e configurar cada micro individualmente e ainda assim ter que conviver com um sistema limitado e lento, é possível transformar micros a partir de 486 em terminais leves, onde um servidor mais rápido executa os aplicativos e envia apenas a imagem a ser mostrada na tela para os terminais. Isto é feito usando uma combinação de servidor de arquivos e servidor de acesso remoto, instalada através do LTSP” [Morimoto (2005)].

Então, nota-se que a utilização da ferramenta *LTSP* para construir uma rede de computadores antigos com recursos modernos, facilita bastante pelo fato de as máquinas precisarem apenas de mínimos equipamentos (placa mãe, pouca memória, disquete, placa de vídeo e placa de rede), não necessitando de dispositivos de armazenamento.

4.2 Entendendo o *LTSP*

O *LTSP* é na verdade uma espécie de distribuição GNU/Linux destinada a ser carregada pelos terminais. Ele é composto por um conjunto de pacotes, que criam um sistema de arquivos dentro da pasta “*/opt/ltsp/i386/*”, que é compartilhada com a rede e acessada via *NFS* pelos clientes como se fosse uma partição local.

Dentro do diretório vai um sistema simplificado, destinado apenas a detectar o *hardware* do cliente e permitir que ele abra uma seção do X. Terminado o *boot*, o cliente obtém a tela de *login* do servidor via *XDMCP*. A partir daí, o servidor roda os aplicativos e o cliente apenas mostra as imagens geradas na tela, atuando como uma espécie de terminal burro.

O *LTSP* é carregado nos clientes usando uma série de serviços. Tudo começa com o cliente dando *boot* usando a imagem de *boot* gravada no disquete ou CD. Esta imagem contém um *software* muito simples, que ativa a placa de rede e envia um pacote de *broadcast*, pedindo a configuração da rede. Um servidor *DHCP* instalado no servidor *LTSP* é configurado para responder ao chamado, enviando a configuração da rede, juntamente com informações do *Kernel*, que o cliente deve carregar via *TFTP*, e a pasta no servidor com a instalação do *LTSP*, que deve ser acessada via *NFS*.

O *TFTP* é um protocolo bem simples de transferência de arquivos dentro de redes locais. Tão simples que a imagem de *boot*, com seus poucos *kbytes*, é grande o suficiente para incluir um cliente *TFTP*, usado na etapa inicial de *boot*. Depois que o *Kernel* é carregado via *TFTP*, começa o *boot* real da estação. O *TFTP* é substituído então por um cliente *NFS* (muito mais robusto que o *TFTP*), que é usado para montar a pasta “*/opt/lts/i386*” do servidor (em modo somente leitura) como diretório raiz no terminal. A estação pode então carregar o sistema do *LTSP*, que se encarrega de detectar o *hardware* da estação e abrir o X. Todos os arquivos de configuração e alterações geradas nesta fase são salvos num *ramdisk*, já que a estação não tem permissão para alterar os arquivos do servidor.

Opcionalmente é possível especificar também uma configuração separada para cada estação, especificando o tipo de *mouse* e a resolução de tela, por exemplo. Esta configuração fica armazenada num arquivo de configuração central do *LTSP*, o “*/opt/ltsp/i386/etc/lts.conf*” (armazenado no servidor), que é lido pelas estações durante o processo de *boot*.

Além do *LTSP* propriamente dito, é necessário ter instalado um conjunto de serviços, cuidadosamente configurados, para que tudo funcione em conjunto. Isto faz com que o *LTSP* seja um sistema um pouco trabalhoso de instalar, onde é necessário prestar muita atenção em cada passo da configuração, já que qualquer erro pode fazer com que tudo deixe de funcionar.

O *LTSP* inclui dois utilitários de configuração: o *ltspadmin*, que automatiza partes da instalação e configuração inicial do sistema, e o *ltspcfg*, que é utilizado para alterar a configuração depois de instalado. Como são desenvolvidas para trabalhar em conjunto com muitas distribuições, estas duas ferramentas nem sempre funcionam corretamente, por isso, a configuração do *LTSP* será realizada de forma manual, ou seja, editando manualmente os arquivos de configuração.

4.3 Instalando os Serviços

Antes de iniciar a instalação do *LTSP*, é importante instalar e testar os serviços base usados por ele. Os serviços usados pelo *LTSP*, que precisam estar instalados e ativos no servidor são:

tftpd
dhcpcd, no *Debian* o serviço se chama *dhcp3-server*

nfsd
xmcp

O “*tftpd*” é o serviço *TFTP*, um pacote bem pequeno utilizado para transferir o *Kernel* usado pelas estações. Instala-se o pacote “*tftpd*” através dos gerenciadores de pacotes da distribuição usada. Em algumas distribuições o *tftpd* é executado como um serviço, controlado pelo script “*/etc/init.d/tftpd*” e, em outras através do *inetd*.

O segundo pacote necessário é o servidor *DHCP*. No *Debian*, este pacote chama-se “*dhcp3-server*”, e em outras distribuições chama-se apenas “*dhcpd*”.

O *LTSP* utiliza o servidor *NFS* para compartilhar a pasta “*/opt/ltsp/i386*”, que é montada como diretório raiz (*/*) pelos terminais na hora do *boot*. Instala-se aqui, o *Portmap* e o servidor *NFS*.

Finalmente, precisa-se habilitar o *XDMCP*, que permitirá que os terminais obtenham a tela de *login* do servidor e executem os aplicativos remotamente. O *XDMCP* é a parte mais importante ao instalar o *LTSP*. Naturalmente, o suporte a *XDMCP*, que permite às outras máquinas da rede rodar aplicativos instalados no servidor, vem desativado por padrão. O *XDMCP* não utiliza nenhum tipo de encriptação ou compressão, simplesmente transmite os dados da forma mais simples e rápida possível. Loga-se no servidor, carrega-se uma interface gráfica e roda todos os aplicativos instalados no servidor, de uma forma completamente transparente e com um bom desempenho.

O *XDMCP* deve ser utilizado apenas dentro de redes locais. É bom

frisar mais uma vez, que um *firewall* para bloquear conexões provenientes da Internet será bem vindo.

Para configurar o servidor para aceitar as conexões, será necessário alterar mais dois arquivos. O primeiro faz com que o X aceite conexões remotas e o segundo configura o gerenciador de *login* (*KDM* ou *GDM*) para escutar estas conexões e fornecer a tela de *login* aos clientes. Nas distribuições que utilizam o *KDM*, procura-se pelos arquivos *kdmrc* e *Xaccess*. Em algumas distribuições eles ficam em “*/usr/share/config/kdm*” e no *Debian* ficam na pasta “*/etc/kde3/kdm*”.

Dentro do arquivo “*Xaccess*”, descomenta-se a linha:

```
# * #any host can get a login window
```

Basta retirar a tralha (#) e deixando o asterisco como primeiro caractere.

Esta linha faz com que o servidor passe a aceitar conexões de todos os *hosts* da rede. Caso prefira-se limitar o acesso a apenas alguns endereços, basta substituir o asterisco pelos endereços desejados.

Em seguida, edita-se o arquivo *kdmrc*. Quase no final do arquivo encontra-se a seguinte linha:

```
[Xdmcp]  
Enable=false
```

Basta alterá-la para:

```
[Xdmcp]  
Enable=true
```

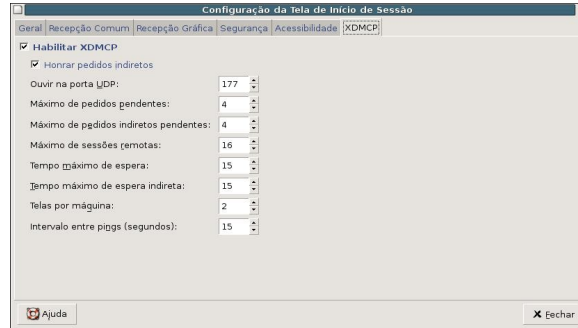


Figura 4.2: Aplicativo *gdmsetup*

No *Debian* existe um programa chamado “*gdmsetup*” para fazer as configurações do *XDMCP* (ressalta-se que aqui o *GDM* é o gerenciador de *login default*). Como mostra a figura 4.2, para fazer com que o *XDMCP* aceite conexões externas, executa-se o programa “*gdmsetup*”, clica na aba “*XDMCP*” e marca a opção “*Habilitar XDMCP*”. É necessário reiniciar o serviço *GDM*.

```
# /etc/init.d/gdm restart
```

O *Debian* (e muitas distribuições derivadas dele) vem com a porta *default* do *XDMCP* bloqueada, uma medida de segurança. Ao utilizar o servidor, será preciso desbloquear a porta editando o arquivo “*/etc/X11/xinit/xserverrc*”, comentando a linha:

```
exec /usr/bin/X11/X -dpi 100 -nolisten tcp
```

4.4 Pacotes do *LTSP*

Até a versão 3.0, estavam disponíveis para *download* um conjunto de pacotes para várias distribuições, incluindo pacotes *.rpm* (*Red Hat*), *.deb* (*Debian*), e *.tgz* (*Slackware*).

Para instalar o *LTSP* 3.0 no *Debian*, por exemplo, baixa-se os seguintes pacotes e utiliza-se o comando “`dpkg -i`” para instalá-los:

```
ltsp-core-i386_3.0.7-3_all.deb  
ltsp-kernel-2.4.19-i386_3.0.5-0_all.deb  
ltsp-x-core-i386_3.0.4-0_all.deb  
ltsp-x-fonts-i386_3.0.0-0_all.deb  
ltsp-x-xserver-fbdev-3.3.6-i386_3.0.0-0_all.deb  
ltsp-x-xserver-svga-3.3.6-i386_3.0.0-0_all.deb
```

Estes pacotes ainda estão disponíveis. Mas, a partir da versão 4.0, foi desenvolvido um sistema unificado de instalação, onde baixa um arquivo *.ISO*, contendo todos os pacotes e um instalador que se encarrega de descompactar os arquivos e ajudar na configuração. Baixa-se a versão mais nova do CD em <http://www.ltsp.org/download/>.

A versão utilizada para o projeto foi a 4.1. A estrutura do *LTSP* não muda muito de uma versão para outra, por isso, mesmo que se esteja instalando uma versão mais recente, a possibilidade dos passos para a instalação serão idênticos.

Pode-se ainda montar o *ISO* numa pasta e instalar diretamente a partir dela, sem a necessidade de queimar um CD:

```
# mount -o loop ltsp-4.1.1-.iso /mnt/cdrom
```

Copia-se o arquivo instalador dentro do CD para uma pasta externa e inicia a instalação:

```
# cp /mnt/cdrom/ltsp-utils-0.11.tgz /tmp  
# cd /tmp  
# tar -zxvf ltsp-utils-0.11.tgz
```

```
# cd ltsp-utils
# ./install.sh
```

Depois de instalado, chama-se o comando “*ltspadmin*” para abrir o instalador. Ele é escrito em *Perl* e precisa que o pacote “*libwww-perl*” esteja instalado.

```
# ltspadmin
```

Seleciona-se a segunda opção “*Configure the installer options*”. Aqui define a localização dos pacotes e quais serão instalados. Por padrão, o instalador se oferece para baixar os pacotes via *web*. Mas, como a instalação aqui será realizada a partir do *ISO*, responde-se “*file:///mnt/cdrom*” no “*Where to retrieve packages from?*”. Ressalta-se que a localização contém três barras, pois é a junção de “*file:///*” e a pasta onde o *ISO* ou CD-ROM está acessível (*/mnt/cdrom*). O diretório de instalação é “*/opt/ltsp*”.

O instalador se oferece para configurar um *proxy*, porém isso só será necessário ao instalar via *web*. No final, responde-se “*y*” para salvar a configuração, conforme figura 4.3.

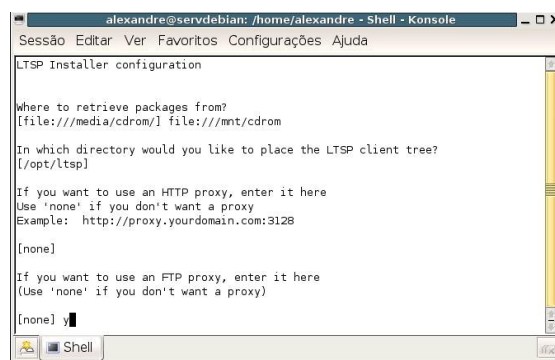


Figura 4.3: Aplicativo *ltspadmin*

De volta à tela inicial, escolhe-se agora a primeira opção “*Install/Update LTSP Packages*”. Será apresentada a tela de seleção dos pacotes a instalar (figura 4.4). Pressiona-se aqui a tecla “A” para marcar todos os pacotes e “Q” para iniciar a instalação.

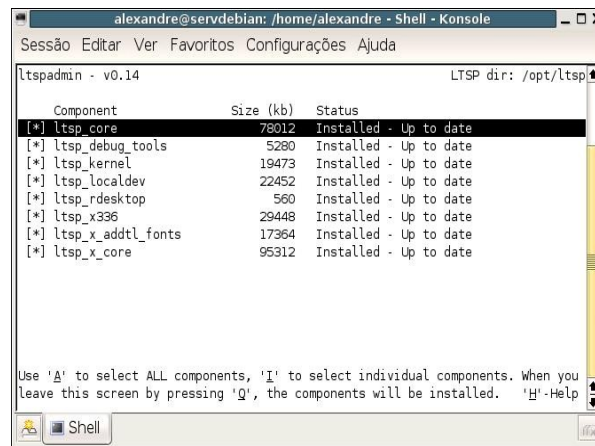


Figura 4.4: *ltspadmin* Escolhendo os Aplicativos

Depois de concluída a instalação é interessante acessar a pasta “*/opt/ltsp/i386*”, e notar que ela foi povoada com um conjunto de pastas padrão encontrado numa distribuição GNU/Linux, como mostra a figura 4.5.

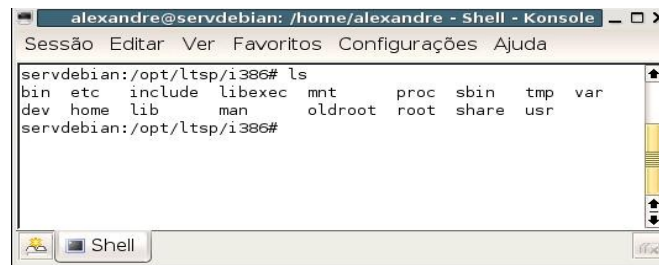


Figura 4.5: Diretório */opt/ltsp/i386*

A instalação do *LTSP* é bem simplificada, principalmente usando o novo instalador. A parte mais complexa é a configuração dos serviços-base e a configuração das estações, que fica no arquivo “*/opt/ltsp/i386/etc/lts.conf*”.

Finalizada a instalação, tem a opção de utilizar a terceira opção do instalador: “*Configure LTSP*”, mostrada na figura 4.6, que verifica se os serviços necessários estão habilitados e ajuda dando a opção de criar um arquivo de configuração padrão para cada um. De qualquer forma, a maior parte da configuração precisa ser feita manualmente, ele só ajuda criando os modelos de arquivo e verificando alguns problemas comuns.

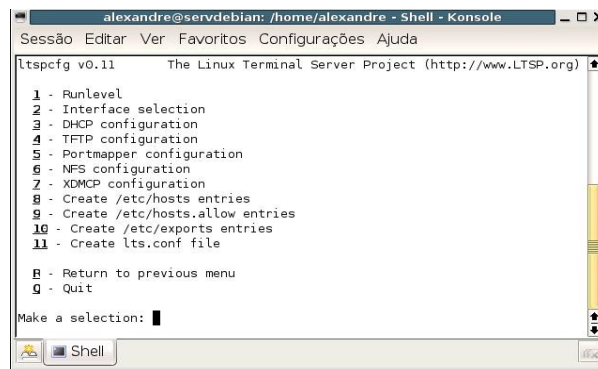


Figura 4.6: Configurações no *ltspadmin*

Para ajudar na configuração, segue no anexo A todos os seis arquivos utilizados no *LTSP*. São eles:

- dhcpcd.conf*** (a configuração do servidor *DHCP*)
- exports*** (a configuração do servidor *NFS*)
- hosts*** (onde vão os endereços IP e nomes das estações)
- hosts.allow*** (permissões de acesso)
- inetd.conf*** (a configuração *inetd*, responsável por carregar o *tftpd*)
- lts.conf*** (onde vai a configuração de cada estação)

4.5 Configuração dos Terminais

A primeira coisa a fazer é acessar a página da rom-o-matic³, para criar os disquetes ou CDs de *boot* necessários para as estações iniciarem corretamente. É necessário indicar apenas o modelo da placa de rede e gravar a imagem seguindo as recomendações da página. No primeiro campo, indica-se o modelo da placa de rede e, no segundo, o tipo de imagem que será gerada. Há módulos para várias placas de rede: *3Com*, *Intel*, *sis900*, *via-rhine*, *realtek8139*, etc.

Utiliza-se a opção “*Floppy Bootable Rom Image (.zdisk)*” para gerar a imagem de um disquete de *boot* ou a opção “*ISO Bootable Image With Legacy Floppy Emulation (.liso)*” para gerar um CD de *boot*. Neste caso, modifica o arquivo gerado de “*.liso*” para “*.iso*”. Em caso de dúvidas sobre os módulos usados pelas placas, pode-se consultar a tabela em: <http://www.etherboot.org/db/>.

3 - <http://www.rom-o-matic.org/>

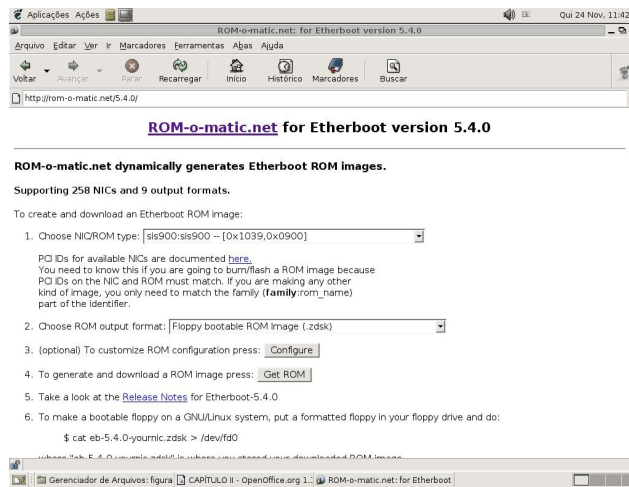


Figura 4.7: Página Oficial do Projeto Rom-o-Matic

Para gravar um CD de *boot*, basta gravar a imagem de forma como gravaria um arquivo .iso tradicional. Depois, configura-se o cliente par dar *boot* através do CD-ROM. A opção “*With Legacy Floppy Emulation*”, torna a imagem compatível com os *BIOS* usadas em micros antigos.

Para gravar os disquetes, usa-se o comando:

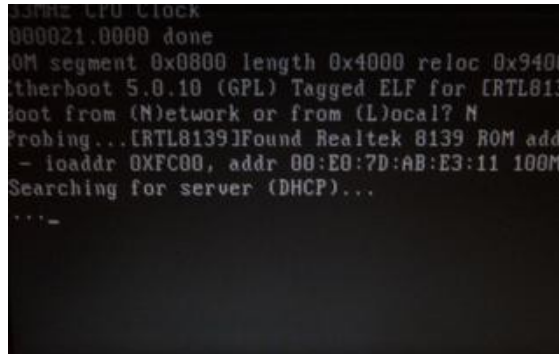
```
# dd if=eb-5.0.10-rtl8139.lzdsk of=/def/fd0
```

(onde eb-5.0.10-rtl8139.lzdsk é o nome do arquivo do módulo da placa)

Após a criação dos discos, dá-se um *boot* usando o disquete ou CD apropriado em cada estação e anota o número do endereço *MAC* de cada placa de rede, que é mostrado no início do *boot*. Será necessário fornecer os endereços *MAC* de cada estação nos arquivos de configuração do *LTSP*.

O endereço *MAC* é um número de 12 dígitos (como: 00:0C:29:6F:F4:AB), diferente em cada placa. Ele aparece na penúltima linha,

conforme mostra a figura 4.8.

A terminal window showing the boot process. The text is as follows:

```
000021.0000 done
ROM segment 0x0800 length 0x4000 reloc 0x9400
etherboot 5.0.10 (GPL) Tagged ELF for [RTL813
boot from (N)etwork or from (L)ocal? N
Probing...[RTL8139]Found Realtek 8139 ROM add
- loadaddr 0XFC00, addr 00:E0:7D:AB:E3:11 100MB
Searching for server (DHCP)...
```

Figura 4.8: Processo de *Boot* no Terminal

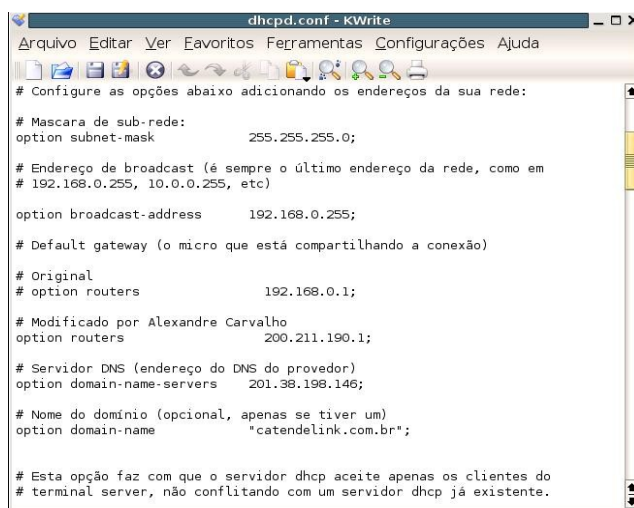
Enquanto o servidor não estiver configurado, os terminais vão ficar indefinidamente no “*Searching for DHCP Server...*”, um sinal de que o disquete conseguiu ativar corretamente a placa de rede.

4.6 Configuração do Servidor

Nesta etapa vem a parte mais complexa, que é a configuração propriamente dita, realizada em seis arquivos distintos. É bom frisar que em qualquer arquivo de configuração as linhas iniciadas por um “#” são consideradas comentários que não possuem efeito algum.

A primeira parada é o arquivo “*/etc/dhcp3/dhcpd.conf*” (Figura 4.9). Aqui fica a configuração do servidor *DHCP*, que diz aos clientes qual é o endereço IP de cada um e qual o endereço IP e pasta do servidor que contém o *Kernel* que será usado na etapa inicial do *boot*.

Na primeira parte do arquivo é necessário fornecer as configurações da rede, como a máscara de sub-rede, o endereço do *gateway* e o *DNS* do provedor. Utilizou-se no projeto a faixa de IP 192.168.0.x, onde o servidor de terminais é configurado para usar o endereço 192.168.0.1. Já os terminais dispõem de endereços IP entre 192.168.0.11 a 192.168.0.16. Ressalta-se que foram utilizados seis terminais.



```
# Configure as opções abaixo adicionando os endereços da sua rede:
# Mascara de sub-rede:
option subnet-mask 255.255.255.0;

# Endereço de broadcast (é sempre o último endereço da rede, como em
# 192.168.0.255, 10.0.0.255, etc)
option broadcast-address 192.168.0.255;

# Default gateway (o micro que está compartilhando a conexão)
# Original
# option routers 192.168.0.1;
# Modificado por Alexandre Carvalho
option routers 200.211.190.1;

# Servidor DNS (endereço do DNS do provedor)
option domain-name-servers 201.38.198.146;

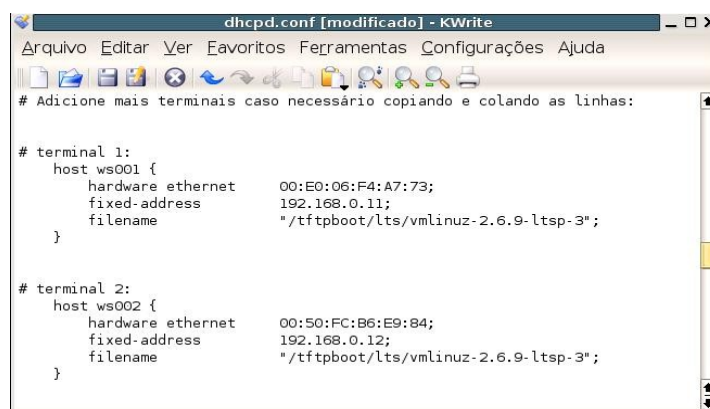
# Nome do domínio (opcional, apenas se tiver um)
option domain-name "catendelink.com.br";

# Esta opção faz com que o servidor dhcp aceite apenas os clientes do
# terminal server, não conflitndo com um servidor dhcp já existente.
```

Figura 4.9: Arquivo `/etc/dhcp3/dhcpd.conf` – Parte 1

A seguir vem a opção onde deve fornecer o endereço IP usado pelo servidor *LTSP*. O `"/etc/ltsp/i386"` representa a pasta de instalação do *LTSP*, que é montada pelos clientes como diretório raiz durante o *boot*. Não se deve esquecer de verificar e alterar esta configuração se tiver instalado o *LTSP* em outra pasta ou estiver utilizando outro endereço IP no servidor (Figura 4.10).

Podem ser configuradas duas estações para carregar arquivos de *Kernel* distintos, tudo de forma independente, ou seja, o servidor é capaz de direcioná-la pelo endereço *MAC* da placa de rede (Figura 4.11).



```
dhcpd.conf [modificado] - KWrite
Arquivo  Editar  Ver  Favoritos  Ferramentas  Configurações  Ajuda
# Adicione mais terminais caso necessário copiando e colando as linhas:

# terminal 1:
host ws001 {
    hardware ethernet    00:E0:06:F4:A7:73;
    fixed-address        192.168.0.11;
    filename              "/tftpboot/lts/vmlinuz-2.6.9-1tsp-3";
}

# terminal 2:
host ws002 {
    hardware ethernet    00:50:FC:B6:E9:84;
    fixed-address        192.168.0.12;
    filename              "/tftpboot/lts/vmlinuz-2.6.9-1tsp-3";
}
```

Figura 4.11: Arquivo */etc/dhcp3/dhcpd.conf* – Parte 3

O próximo arquivo é o “*/etc/exports*” (Figura 4.12), onde fica a configuração do servidor *NFS*. O *LTSP* utiliza dois compartilhamentos: o diretório “*/opt/ltsp/i386*” (o compartilhamento principal, que contém o sistema de arquivos utilizado pelas estações), e uma pasta vazia “*/var/opt/ltsp/swapfiles*” (onde ficam os arquivos de *swap* usados pelas estações).

Um exemplo de configuração do arquivo */etc/exports*:

```
/opt/ltsp/i386          192.168.0.0/255.255.255.0(ro,no_root)squash)
/var/opt/ltsp/swapfiles 192.168.0.0/255.255.255.0(rw,no_root_squash)
```

Isto é opcional. Este recurso permite que estações desprovidas de *HD* utilizem o *HD* do servidor para fazer a *swap*. Só há realmente necessidade no caso de micros antigos, com menos de 32 MB de *RAM*. Deve ser criada a pasta antes de compartilhar o diretório de *swap*.

```
# mkdir -p /var/opt/ltsp/swapfiles
```

Já o arquivo “*/etc/hosts.allow*” (Figura 4.13), determina quais micros da rede terão acesso a cada serviço do sistema. No caso do *LTSP*, utiliza-se um conjunto de serviços que transmitem dados sem encriptação, como o *TFTP*, *XDMCP* e o próprio *NFS*, por isso é importante que o servidor seja configurado para não receber conexões vindas da Internet.

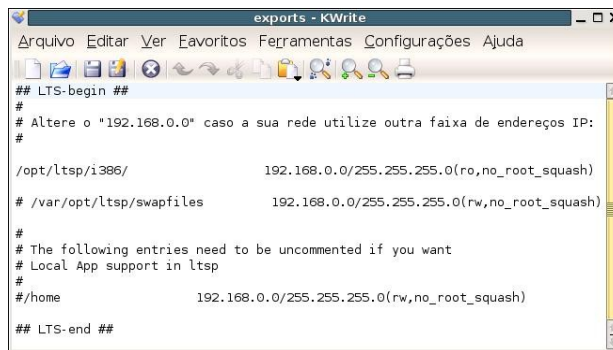
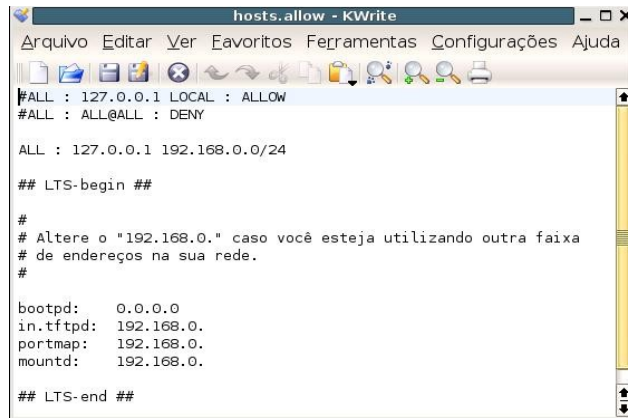


Figura 4.12: Arquivo */etc/exports*

O ideal aqui, é que se utilize um servidor separado para o *LTSP*, que acessa a Internet através de uma conexão compartilhada por outro servidor, ou pelo menos ativar um *firewall*, permitindo apenas conexões provenientes da rede local.



```
#ALL : 127.0.0.1 LOCAL : ALLOW
#ALL : ALL@ALL : DENY

ALL : 127.0.0.1 192.168.0.0/24

## LTS-begin ##

#
# Altere o "192.168.0." caso você esteja utilizando outra faixa
# de endereços na sua rede.
#

bootpd: 0.0.0.0
in.tftpd: 192.168.0.
portmap: 192.168.0.
mountd: 192.168.0.

## LTS-end ##
```

Figura 4.13: Arquivo */etc/hosts.allow*

No *Debian* (e derivados), há necessidade de ajustar o arquivo *“/etc/inetd.conf”*, onde deve ser incluída a linha que inicializa o serviço *TFTP*:

```
tftp dgram udp wait nobody /usr/sbin/tcpd /usr/sbin/in.tftpd /tftpboot
```

Em suma, esta linha compartilha a pasta *“/tftpboot”* (onde está a imagem do *Kernel*) usando o *tftpd*. Depois que o arquivo é salvo, deve-se reiniciar o serviço *inetd* para que a configuração entre em vigor:

```
# /etc/init.d/inetd restart
```

O *tftpd* tem uma função essencial logo no início do *boot* das estações. Caso ele não esteja corretamente ativado, nada funcionará.

Finalmente a parte mais importante da configuração, o arquivo *“/opt/ltsp/i386/etc/lts.conf”* (Figura 4.14). É aqui onde ficam armazenadas as configurações de resolução de vídeo, qual o tipo de *mouse* de cada estação, e ainda, a opção de ativar ou não a *swap*.

Logo no início deve-se prestar bastante atenção para substituir os dois “192.168.0.1” pelo IP correto do servidor, caso contrário os clientes não conseguirão dar *boot*. Existem dois campos: o “*SERVER*” (o servidor com *LTSP* instalado, que fica responsável por enviar os arquivos de *boot*) e o “*XDM_SERVER*” (o servidor com o *XDMCP* ativo, que realmente roda os aplicativos gráficos). Em geral, uma única máquina cuida das duas funções, mas nada impede que se use dois servidores separados para as duas funções.

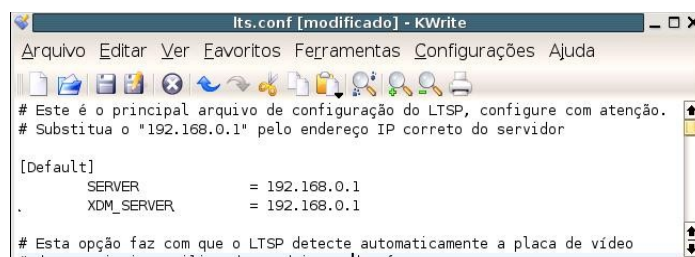


Figura 4.14: Arquivo `/opt/ltsp/i386/etc/its.conf` – Parte 1

Abaixo, a partir da opção “*XSERVER*”, segue a configuração padrão do *LTSP*. Não existe necessidade de modificar nada aqui, pois pode especificar configurações diferentes para cada estação mais abaixo, especificando diferentes resoluções de vídeo, tipos de *mouse* e taxas de atualização de monitor. É bom lembrar que as estações executam localmente uma cópia do *Kernel*, utilitários básicos e uma instância do X. Graças a isso, a configuração de vídeo de cada estação é completamente independente do servidor. Nada impede que uma estação use um monitor de 17” a 1280x1024, enquanto outra usa um *VGA Mono* a 640x480.

Este é o principal motivo de relacionar os endereços *MAC* de cada placa de rede com um nome de terminal e endereço IP específico na configuração do *DHCP*. Graças a isso, o servidor consegue diferenciar os terminais e enviar a

configuração correta para cada um (Figura 4.15).

Figura 4.15: Arquivo `/opt/ltsp/i386/etc/ltsp.conf` – Parte 2

O *LTSP* 3.0 utiliza o *Xfree* 4, enquanto a versão 4.1 já utiliza o *X.org*. Ambos possuem um sistema de detecção automática para o vídeo em cada estação (a opção “*XSERVER = auto*”). Ao final do *boot* ele tentará detectar a placa de vídeo e detectar as taxas de atualização suportadas pelo monitor. Este sistema funciona direto em uns dois terços dos micros, mas em um grande número de casos precisa-se especificar algumas configurações manualmente para que tudo funcione corretamente.

A configuração padrão do *LTSP* é usar um *mouse PS/2* (sem roda) em todas as estações. Evidentemente, existirão alguns micros com *mouse serial* ou *PS/2* com roda, o que também há necessidade de se ajustar. Esta configuração individual das estações de trabalho é feita logo abaixo, relacionando o nome de cada estação com as opções desejadas.

A configuração utilizada por padrão especifica pouca coisa para as estações. Neste caso, pode-se começar testando cada estação com esta

44

configuração padrão e depois se concentrar em configurar manualmente apenas aquelas em que o X não abriu, onde a resolução detectada não seja a desejada:

```
[ws001]
XSERVER           = auto
USE_NFS_SWAP      = Y
SWAPFILE_SIZE     = 32m
RUNLEVEL          = 5
```

O “*XSERVER = auto*”, faz com que o *LTSP* detecte automaticamente a placa de vídeo em cada estação. O “*USE_NFS_SWAP = Y*” ativa o recurso de *swap* via rede para as estações. Já o “*SWAPFILE_SIZE = 32m*” especifica o tamanho do arquivo de *swap*, enquanto que o “*RUNLEVEL = 5*” faz com que as estações dêem *boot* direto em modo gráfico. Caso queira alguma estação trabalhando em modo texto, basta modificar o 5 por 3. Note que ao usar estações com 32 MB de RAM ou mais, pode-se desativar o recurso de *swap* via rede com segurança.

Além destas, existem várias outras opções que podem ser usadas. Se a detecção automática do vídeo não funcionar (a tela vai piscar algumas vezes e depois voltar ao modo texto), pode-se indicar manualmente um *driver*, substituindo o “*auto*” por “*vesa*” (um *driver* genérico, um pouco mais lento, mas que funciona na maioria das placas). Outros *drivers* disponíveis são: *cirrus*, *cyrix*, *i810*, *neomagic*, *r128*, *radeon*, *rendition*, *s3virge*, *sis*, *tdfx* e *trident*. Pode-se ainda encontrar detalhes sobre as placas suportadas por cada um no *site*: <http://www.x.org/X11R6.8.2/doc/>

Este é um exemplo de configuração de um *Pentium 100*, com uma placa de vídeo *Trident 9680*, ligado num monitor de 14”:

```
[ws001]
XSERVER                = XF86_SVGA
X_MODE_0               = 800x600
X_HORZSYNC             = 30-54
X_VERTREFRESH          = 50-72
X_COLOR_DEPTH          = 16
RUNLEVEL               = 5
```

Exemplo para usar um *mouse serial* na estação:

```
[ws001]
XSERVER                = auto
X_MOUSE_PROTOCOL      = "Microsoft"
X_MOUSE_DEVICE        = "/dev/ttyS0"
X_MOUSE_RESOLUTION    = 400
X_MOUSE_BUTTONS       = 2
X_MOUSE_EMULATE3BTN   = Y
RUNLEVEL               = 5
```

Exemplo para usar um *mouse PS/2* com roda na estação:

```
[ws001]
XSERVER                = auto
X_MOUSE_PROTOCOL      = "IMPS/2"
X_MOUSE_DEVICE        = "/dev/psaux"
X_MOUSE_RESOLUTION    = 400
X_MOUSE_BUTTONS       = 5
X_ZaxisMapping        = "4 5"
RUNLEVEL               = 5
```

O exemplo abaixo força a estação de trabalho a usar uma configuração de resolução e taxa de atualização específica para o monitor. Ele é útil em casos em que o X chega a abrir, mas o monitor fica fora de sincronia. Isso acontece em muitos micros antigos, onde o monitor ou a placa de vídeo não são compatíveis com o protocolo *DCC*:

[ws001]

XSERVER	= auto
X_MODE_0	= 1024x768
X_VERTREFRESH	= 60
X_COLOR_DEPTH	= 16
RUNLEVEL	= 5

Outra configuração que pode ser importante é o teclado. Por padrão, o *LTSP* vem configurado para usar um teclado padrão americano, sem acentuação. Ao carregar o *KDE*, passam a valer as configurações do *Kxkb*, o gerenciador de teclado do *KDE*, da forma como configuradas no *Painel de Controle do KDE*. Mas, esta configuração manual do *LTSP* é necessária ao utilizar outras interfaces gráficas.

Para um teclado *ABNT2*, inclui-se na configuração da estação as linhas:

XkbModel	= ABNT2
XkbLayout	= br

Para um teclado *Us Internacional*:

XkbModel	= pc104
XkbLayout	= us_intl

A última configuração é o arquivo “*etc/hosts*” (Figura 4.16), onde ficam relacionados os endereços IP e os nomes de cada estação (*ws001*, *ws002*, etc.).

```
hosts - KWrite
Arquivo  Editar  Ver  Favoritos  Ferramentas  Configurações  Ajuda
127.0.0.1 kurumin localhost

# The following lines are desirable for IPv6 capable hosts
# (added automatically by netbase upgrade)

::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts

# Você pode adicionar aqui os endereços IP e os nomes
# correspondentes de cada terminal, caso queira utilizar
# mais de 5 terminais.

# IMPORTANTE: A primeira linha deve conter o endereço IP
# e o nome (definido durante a configuração da rede) do
# servidor, ou seja, desta máquina. Se o nome for diferente
# do definido na configuração da rede, as estações não
# conseguirão montar o sistema de arquivos do LTSP via NFS
# e travarão no boot.

192.168.0.1 servdebian
192.168.0.11 ws001
192.168.0.12 ws002
192.168.0.13 ws003
192.168.0.14 ws004
192.168.0.15 ws005
192.168.0.16 ws006
```

Figura 4.16: Arquivo `/etc/hosts`

Caso tudo esteja ok, depois de reiniciar o servidor, os terminais já conseguirão carregar o sistema e pegar a tela de *login* do servidor.

A partir daí, já pode ir criando as contas de usuários que irão utilizar os terminais. A conexão com a Internet, impressora, disquete e gravador instalados no servidor podem ser usados em qualquer um dos terminais, pois os programas estão realmente no servidor: os terminais funcionam como se fossem vários monitores e teclados ligados a ele. Logicamente, existe muito mais do que foi abordado aqui, o objetivo do projeto é dar um início aos estudos, qualquer um pode melhorar e muito os recursos disponíveis pelo *LTSP*, tais como: *boot* via placa de rede; usar dispositivos locais como *drives* de disquete e CD-ROM; configuração de placas de som; impressoras nos próprios terminais.

A Figura 4.17 mostra a tela de login do GDM (padrão do Gnome) esperando pelo acesso do usuário.



Figura 4.17: Tela de *Login* do GDM

4.7 Alguns Problemas

O primeiro problema encontrado na configuração do *LTSP* foi o serviço *ftpd*, que na primeira instalação não funcionou. Depois de realizada uma nova instalação o serviço funcionou adequadamente. Pode ter sido alguma biblioteca dependente que não tenha sido instalada na primeira tentativa.

O *LTSP 4.1* utiliza o *X.org* como servidor gráfico. O *Debian* utiliza o *Xfree*. Isto resultará num problema com a configuração do teclado. O *Xkb*, o programa responsável por ler e ativar o *layout* de teclado não funciona nesta configuração. O erro acontece por que os clientes carregam o *X.org* incluído no

LTSP e o utilizam par obter uma seção *XDMCP* do sistema instalado no servidor, que roda o *Xfree*. A solução para este erro é instalar o *X.org* no servidor ou passar a utilizar o *LTSP 4.0* que ainda usa o *Xfree*. Para fazer a instalação do *X.org* no *Debian* foi utilizado o tutorial localizado no site do Guia do Hardware⁴.

Na Figura 4.18 é mostrado um terminal sem HD, apenas com drive de disquete, placa de vídeo e placa de rede.



Figura4.18: Terminal sem HD

4 Guia do Hardware (<http://www.guiadohardware.net/linux/dicas/104.htm>)

CAPÍTULO V

EXPERIÊNCIA EM SALA DE AULA



Figura 5.1: Terminal rodando o *OpenOffice Writer*

5.1 A Sala de Aula

Como já mencionado na introdução deste trabalho a execução deste projeto predispõe de seis terminais para a prática do projeto em uma sala de aula na empresa CETAP. As atividades tiveram início em outubro de 2005 e não houve custos para os alunos. A partir de 2006, a empresa estabelecerá uma taxa mínima apenas para a manutenção do módulo, e também para cobrir as despesas ocasionadas por ele.

5.2 O Início do Projeto LTSP e Inclusão Digital no CETAP

Num país onde há muitos excluídos digitais e, principalmente, excluídos

sociais (um grande problema na América Latina), precisa-se ter um cuidado redobrado na hora em que o aluno terá o seu primeiro contato com a micro-informática. Ele vem de um mundo completamente diferente, e terá de se adaptar as novas técnicas (principalmente os termos), como *hardware*, *software*, sistema operacional, aplicativos, entre outros. Então, o curso inicia com as lições preliminares de informática para que haja um completo entendimento do aluno.

5.3 Introdução ao Processamento de Dados

Nesta etapa do aprendizado, o aluno terá seu foco de aprendizado nos componentes que formam um computador. São eles: teclado, *mouse*, cpu, monitor e impressora. Depois entenderá como funcionam os equipamentos de entrada e saída, as unidades de medida como *byte*, *kbyte*, *mbyte*, etc. Passada esta fase, o aprendiz se relacionará com mais aprofundamento no processamento de dados. Ele precisa conhecer as mídias, ou mais precisamente, as unidades de armazenamento (disquete, CD, HD), para poder ter uma noção mais acurada de arquivamento.

5.4 Debian GNU/Linux Sarge 3.1

O *Debian GNU/Linux Sarge 3.1*, é uma excelente distribuição, e para o autor, uma das mais sérias. Encontra-se tudo o que se precisa de *software* em uma distribuição como esta (verdade que muitas distribuições são baseadas em *Debian*).

O aluno terá como aprendizado os gerenciadores de ambiente *Gnome* e *KDE*, tendo em vista a sua facilidade e também o grande número de aplicações que já vem por padrão em cada um desses ambientes.

Entenderá as filosofias de arquivos e pastas, conhecerá o diretório */home*, modo protegido, acesso a disquetes e CDs, e ainda, alguns programas simples para digitar texto, fazer cálculos, abrir fotos e imagens, etc.

Para concluir o módulo de sistema operacional, conhecerá os principais gerenciadores de arquivos, *konqueror* e *nautilus*.

5.5 OpenOffice.org

O *OpenOffice.org* é uma pacote de escritório que reúne vários programas, dentro os quais destacam-se:

Writer – Processador de Textos;
Calc – Planilha Eletrônica;
Impress – Gerenciador de Apresentações;
Draw – Criação e Edição de Desenhos Vetoriais;
Math – Construtor de Fórmulas Matemáticas;
Base – Construtor de Banco de Dados;
Documento HTML – Editor de HTML.

Para este curso foram oferecidos o Writer e o Calc.

***Writer* – Processador e Editor de Textos**

O *OpenOffice.org Writer* é um programa de criação e edição de textos, sendo portanto uma ferramenta indispensável para que o aluno inicie as atividades com o computador. O mesmo poderá ser usado para produzir documentos simples com uma carta, currículos, memorandos, recibos, ou até documentos mais complexos.

O aluno aprenderá a redigir textos com uma digitação correta, para ter uma total qualificação no mercado de trabalho. É fundamental que o aprendiz conheça as regras de digitação perfeitamente.

***Calc* – Planilha Eletrônica**

O *Calc* é um programa que trabalha com planilhas. Neste módulo, o aluno irá ver alguns recursos básicos deste poderoso programa do pacote *OpenOffice.org*.

Aqui serão realizadas tarefas de álgebras comuns (adição, subtração, multiplicação e divisão), para que se tenha uma idéia de como uma planilha funciona. Depois, aprenderá como formatar planilhas, colocar grades, e outras coisas mais. Realizado este processo, o aluno verá itens mais avançados como: média, juros, etc.

5.6 Navegador Internet *Mozilla*

Para navegar na Internet utilizou-se o *software Mozilla*, um ótimo *browser* com diversos recursos e bastante leve.

Aqui, o pupilo terá o seu primeiro contato com a *web*, fará inúmeras visitas a *sites*, principalmente os de cunho educativo. Realizará tarefas como pesquisas em *sites* de busca, baixará arquivos da *net* (*downloads*) para o computador e, por fim, criará uma conta de *e-mail* num provedor gratuito.

5.7 Tarefa Realizada

Ao final de tudo, a empresa emitirá um certificado de participação do aluno no curso básico em *Software Livre* (*GNU/Linux*, *OpenOffice* e Internet), garantindo assim, o ingresso dos participantes na “Inclusão Digital”, e ainda, ficando a empresa com uma imensa satisfação em poder colocar pessoas que não dispõem de recursos no mundo digital.

Conclui-se que, não é difícil promover a Inclusão Digital, muito menos angariar recursos para a sua realização, o maior obstáculo é o ato de vontade. As comunidades, os poderes públicos e os entes privados devem dar as mãos para trabalhar por um Brasil melhor e mais digno.

CONCLUSÕES

O *Linux Terminal Server Project*, é uma tecnologia que permite aos Pcs serem executados em rede e tendo sua inicialização a partir de disquetes ou CD não precisando de unidades de armazenamento. Há apenas a necessidade de um mecanismo para inicialização das máquinas, como por exemplo: disquete ou CD-ROM. Com isto não tendo necessidade de se usar máquinas potentes. Os terminais podem ser micros bem antigos (a partir de um 80486), e terão performance de computador atual.

É uma solução bastante viável para promover a inclusão digital, tendo em vista os seus reduzidos custos com aquisição de *hardware*. Sem dúvida, a melhor forma de fazer centros de computação para as comunidades. Além do mais, é bem prático o aumento de máquinas na rede, bastando apenas fazer a inclusão destas nos arquivos de configuração e reiniciar o serviço *dhcpcd*, sem necessitar de boot no servidor. A manutenção é totalmente centralizada, os aplicativos são instalados / atualizados apenas no servidor e os terminais já terão os mesmos em pleno funcionamento.

Em se tratando do trabalho em sala de aula, a ferramenta *LTSP* teve um rendimento considerável, posto que praticamente não houve problemas técnicos e de *hardware*. O aluno liga o computador com o disquete e pronto, já está no ambiente GNU/Linux para fazer as atividades. O processo de desligamento também ficou facilitado, requer apenas um *logout* e depois o desligamento do computador.

O projeto está em pleno funcionamento na CETAP – Consultoria e Informática, localizada na cidade de Catende/PE. Há uma perspectiva para o aumento de máquinas no início de 2006, e também, inúmeras implementações, tais como: configuração de unidades de CD-ROM locais nos terminais; instalação de um impressora em um dos terminais; configuração de placas de som.

Em suma, o uso da tecnologia LTSP apresentou-se proveitoso, tanto para quem está administrando (instrutor) quanto para quem vive a inclusão digital (aluno). Espera-se que o trabalho tenha algum tipo de benefício para aqueles que têm interesse na aplicação *LTSP*, tornando-a uma ferramenta de uso permanente.

REFERÊNCIAS BIBLIOGRÁFICAS

[Castells (2003)] Castells, Manuel. **A Galáxia da Internet**, Editora: Jorge Zahar, Edição: 1, Publicação: 2003.

[Cruz (2005)] Cruz, Renato. **O Que As Empresas Podem Fazer Pela Inclusão Digital**

Disponível:

<http://www.uniethos.org.br/DesktopDefault.aspx?TabID=3684&Alias=Uniethos&Lang=pt-BR>

Acesso em: 02 de setembro de 2005.

[Silveira (2003)] Silveira, Sérgio Amadeu. **Software Livre e Inclusão Digital**, Editora: Conrad, Edição: 1, Publicação: 2003.

[Levy (1999)] Levy, Pierre. **Cibercultura**, Editora: 34, Edição: 1, Publicação: 1999.

[Raymond (1999)] Raymond, Eric. **A Catedral e o Bazar**

Disponível: <http://www.geocities.com/CollegePark/Union/3590/pt-cathedral-bazaar-1.html>

Acesso em: 05 de setembro de 2005.

[Carpanez (2005)] UOL Folha OnLine – **Caderno de Informática**

Disponível em: <http://www1.folha.uol.com.br/folha/informatica/ult124u18585.shtml>

Acesso em: 20 de novembro de 2005.

Cursos Técnicos no Censo Escolar 2004

Disponível em: http://portal.mec.gov.br/setec/arquivos/pdf/censo_comenta.pdf

Acesso em: 05 de setembro de 2005.

[Morimoto (2005)] Morimoto, Carlos. **Redes e Servidores Linux**, Editora: Sul Editores, Edição: 1, Publicação: Agosto/2005.

[Medida Provisória nº 252, de 15 de junho 2005]

[Takahashi (2005)] **Inclusão Digital no Direito Brasileiro**

Disponível em: <http://www.ciberetica.org.br/trabalhos/anais/17-9-c1-1.pdf>

Acesso em: 10 de setembro de 2005.

[Moore (2001)] **Revolution OS. Direção: J.T.S Moore. Worderview Productions LIC, 2001, 1 videodisco (85 min), son., color.**

ANEXO A (Arquivos de Configuração)

Arquivo */etc/dhcp3/dhcpd.conf*

Mantenha descomentadas as duas linhas abaixo se você for utilizar
placas de rede ISA em algumas das estações:

option option-128 code 128 = string;
option option-129 code 129 = text;

shared-network WORKSTATIONS {
 subnet 192.168.0.0 netmask 255.255.255.0 {

 default-lease-time 21600;
 max-lease-time 21600;

Configure as opções abaixo adicionando os endereços da sua rede:

Mascara de sub-rede:
option subnet-mask 255.255.255.0;

Endereço de broadcast (é sempre o último endereço da rede, como em
192.168.0.255, 10.0.0.255, etc)

option broadcast-address 192.168.0.255;

Default gateway (o micro que está compartilhando a conexão)

Original
option routers 192.168.0.1;

Modificado por Alexandre JLF Carvalho
option routers 200.211.190.1;

Servidor DNS (endereço do DNS do provedor)
option domain-name-servers 201.38.198.146;

Nome do domínio (opcional, apenas se tiver um)
option domain-name "catendelink.com.br";

Esta opção faz com que o servidor dhcp aceite apenas os clientes do
terminal server, não conflitando com um servidor dhcp já existente.

deny unknown-clients;

Caso prefira que o servidor DHCP dê IPs de rede local também para os
demais micros da rede, que não estão cadastrados como terminais, comente
a linha acima e descomente a linha abaixo, informando a faixa de endereços
que será usada pelos clientes que não estejam cadastrados como terminais:

```

# range 192.168.0.100 192.168.0.201;

# IMPORTANT!! Substitua o "192.168.0.10" pelo endereço IP do servidor
# esta máquina, se este endereço estiver errado o LTSP não
# funcionará!

option root-path      "192.168.0.1:/opt/ltsp/i386";
}
}

# -----

group {
    use-host-decl-names    on;
    option log-servers      192.168.200.252;

# Aqui vão as configurações dos terminais, cada terminal deve
# ser configurado com um endereço IP diferente e com o endereço
# MAC de sua placa de rede.

# Para saber o endereço MAC de cada terminal, basta dar um boot
# Com o disquete do rom-o-matic, ele mostrará o endereço MAC
# logo no início do boot.

# Este endereço é único, exclusivo de cada placa de rede, é através
# dele que o servidor sabe qual terminal é qual.

# Adicione mais terminais caso necessário copiando e colando as linhas:

# terminal 1:
host ws001 {
    hardware ethernet    00:E0:06:F4:A7:73;
    fixed-address         192.168.0.11;
    filename              "/tftpboot/lts/vmlinuz-2.6.9-ltsp-3";
}

# terminal 2:
host ws002 {
    hardware ethernet    00:50:FC:B6:E9:84;
    fixed-address         192.168.0.12;
    filename              "/tftpboot/lts/vmlinuz-2.6.9-ltsp-3";
}

# terminal 3:
host ws003 {
    hardware ethernet    00:E0:7D:DB:CE:1E;

```

```

        fixed-address    192.168.0.13;
        filename         "/tftpboot/lts/vmlinuz-2.6.9-ltsp-3";
    }

# terminal 4:
host ws004 {
    hardware ethernet   00:E0:7D:70:DE:8C;
    fixed-address       192.168.0.14;
    filename            "/tftpboot/lts/vmlinuz-2.6.9-ltsp-3";
}

# terminal 5:
host ws005 {
    hardware ethernet   00:02:2A:D7:7F:8D;
    fixed-address       192.168.0.15;
    filename            "/tftpboot/lts/vmlinuz-2.6.9-ltsp-3";
}

# terminal 6:
host ws006 {
    hardware ethernet   00:00:1C:D7:D5:4A;
    fixed-address       192.168.0.16;
    filename            "/tftpboot/lts/vmlinuz-2.6.9-ltsp-3";
}

# Para usar uma placa de rede ISA no terminal, descomente as duas linhas
# abaixo e substitua o "NIC=3c509" pelo módulo da placa de rede usada.
# Não altere o "e4:45:74:68:00:00" este não é um endereço MAC, mas sim
# a string que ativa a linha com o módulo da placa!.

#   option option-128  e4:45:74:68:00:00;
#   option option-129  "NIC=3c509";
# }

# Os terminais abaixo estão configurados para dar boot via via PXE
# (o protocolo de boot suportado por muitas placas de rede, onde basta
# configurar o micro para dar boot através da rede, no setup, ou
# pressionar F12 durante o boot). Substitua a seção referente à estação
# pelo exemplo abaixo. Veja que o que muda é a opção "filename":

# terminal 7:
#   host ws007 {
#       hardware ethernet 00:0C:29:6F:F4:AB;
#       fixed-address 192.168.0.17;
#       filename "/tftpboot/lts/2.4.26-ltsp-3/pxelinux.0";
#   }

# terminal 8:
#   host ws008 {

```

```
#         hardware ethernet 00:11:D8:00:A3:81;
#         fixed-address 192.168.0.18;
#         filename "/tftpboot/lts/2.4.26-ltsp-3/pxelinux.0";
#     }

# terminal 9:
#     host ws009 {
#         hardware ethernet 00:11:D8:03:B3:85;
#         fixed-address 192.168.0.19;
#         filename "/tftpboot/lts/2.4.26-ltsp-3/pxelinux.0";
#     }
}
```

Arquivo */etc/exports*

```
## LTS-begin ##  
#  
# Altere o "192.168.0.0" caso a sua rede utilize outra faixa de endereços IP:  
#  
  
/opt/ltsp/i386/          192.168.0.0/255.255.255.0(ro,no_root_squash)  
  
/var/opt/ltsp/swapfiles 192.168.0.0/255.255.255.0(rw,no_root_squash)  
  
#  
# The following entries need to be uncommented if you want  
# Local App support in ltsp  
#  
#/home                 192.168.0.0/255.255.255.0(rw,no_root_squash)  
  
## LTS-end ##
```

Arquivo */etc/hosts*

```
127.0.0.1      servdebian localhost

# The following lines are desirable for IPv6 capable hosts
# (added automatically by netbase upgrade)

::1   ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts

# Você pode adicionar aqui os endereços IP e os nomes
# correspondentes de cada terminal, caso queira utilizar
# mais de 6 terminais.

# IMPORTANTE: A primeira linha deve conter o endereço IP
# e o nome (definido durante a configuração da rede) do
# servidor, ou seja, desta máquina. Se o nome for diferente
# do definido na configuração da rede, as estações não
# conseguirão montar o sistema de arquivos do LTSP via NFS
# e travarão no boot.

192.168.0.1 servdebian
192.168.0.11 ws001
192.168.0.12 ws002
192.168.0.13 ws003
192.168.0.14 ws004
192.168.0.15 ws005
192.168.0.16 ws006
```

Arquivo */etc/hosts.allow*

```
#ALL : ALL@ALL : DENY
```

```
ALL : 127.0.0.1 192.168.0.0/24
```

```
## LTS-begin ##
```

```
#
```

```
# Altere o "192.168.0." caso você esteja utilizando outra faixa  
# de endereços na sua rede.
```

```
#
```

```
bootpd: 0.0.0.0
```

```
in.tftpd: 192.168.0.
```

```
portmap: 192.168.0.
```

```
moundd: 192.168.0.
```

```
## LTS-end ##
```

Arquivo */etc/inetd.conf*

Não altere este arquivo a menos que saiba o que está fazendo.

```
# /etc/inetd.conf: see inetd(8) for further informations.
#
# Internet server configuration database
#
#
# Lines starting with "#:LABEL:" or "#<off>#" should not
# be changed unless you know what you are doing!
#
# If you want to disable an entry so it isn't touched during
# package updates just comment it out with a single '#' character.
#
# Packages should modify this file by using update-inetd(8)
#
# <service_name> <sock_type> <proto> <flags> <user> <server_path> <args>
#
#:INTERNAL: Internal services
#echo          stream  tcp      nowait  root    internal
#echo          dgram  udp      wait    root    internal
#chargen      stream  tcp      nowait  root    internal
#chargen      dgram  udp      wait    root    internal
#discard      stream  tcp      nowait  root    internal
#discard      dgram  udp      wait    root    internal
#daytime      stream  tcp      nowait  root    internal
#daytime      dgram  udp      wait    root    internal
#time         stream  tcp      nowait  root    internal
#time         dgram  udp      wait    root    internal

#:STANDARD: These are standard services.
#ftp          stream  tcp      nowait  root    /usr/sbin/tcpd  /usr/sbin/in.ftpd
#telnet       stream  tcp      nowait  root    /usr/sbin/tcpd  /usr/sbin/in.telnetd

#:BSD: Shell, login, exec and talk are BSD protocols.
#talk        dgram  udp      wait    root.tty /usr/sbin/tcpd  /usr/sbin/kotalkd
#ntalk       dgram  udp      wait    root.tty /usr/sbin/tcpd  /usr/sbin/ktalkd

#:MAIL: Mail, news and uucp services.

#:INFO: Info services

#:BOOT: Tftp service is provided primarily for booting. Most sites
# run this only on machines acting as "boot servers."
#tftp        dgram  udp      wait    nobody   /usr/sbin/tcpd  /usr/sbin/in.tftpd
/boot
#bootps     dgram  udp      wait    root    /usr/sbin/bootpd bootpd -i -t 120
```

```

#:RPC: RPC based services

#:HAM-RADIO: amateur-radio services

#:OTHER: Other services
#<off># netbios-ssn    stream  tcp    nowait  root    /usr/sbin/tcpd
                /usr/sbin/smbd
#<off># netbios-ns    dgram   udp     wait    root    /usr/sbin/tcpd
                /usr/sbin/nmbd -a
#<off># swat          stream  tcp     nowait.400    root    /usr/sbin/tcpd
                /usr/sbin/swat
#printer stream tcp nowait lp /usr/lib/cups/daemon/cups-lpd cups-lpd
#vboxd  stream tcp    nowait  root    /usr/sbin/tcpd  /usr/sbin/vboxd
#saft   stream tcp    nowait  root    /usr/sbin/tcpd  /usr/sbin/sendfiled
#xtel   stream  tcp    nowait  root    /usr/sbin/tcpd  /usr/sbin/xteld
#<off># https  stream tcp nowait root /usr/sbin/tcpd /usr/sbin/sslwrap -nocert -addr
127.0.0.1 -port 80
#<off># ssmtp  stream tcp nowait root /usr/sbin/tcpd /usr/sbin/sslwrap -nocert -addr
127.0.0.1 -port 25
#<off># nntps  stream tcp nowait root /usr/sbin/tcpd /usr/sbin/sslwrap -nocert -addr
127.0.0.1 -port 119
#<off># telnets  stream tcp nowait root /usr/sbin/tcpd /usr/sbin/sslwrap -nocert -addr
127.0.0.1 -port 23
#<off># imaps   stream tcp nowait root /usr/sbin/tcpd /usr/sbin/sslwrap -nocert -addr
127.0.0.1 -port 143
#<off># ircs    stream tcp nowait root /usr/sbin/tcpd /usr/sbin/sslwrap -nocert -addr
127.0.0.1 -port 194
#<off># pop3s   stream tcp nowait root /usr/sbin/tcpd /usr/sbin/sslwrap -nocert -addr
127.0.0.1 -port 110
#<off># ftps-data  stream tcp nowait root /usr/sbin/tcpd /usr/sbin/sslwrap -nocert
-addr 127.0.0.1 -port 20
#<off># ftps     stream tcp nowait root /usr/sbin/tcpd /usr/sbin/sslwrap -nocert -addr
127.0.0.1 -port 21
#<off># sswat   stream tcp nowait root /usr/sbin/tcpd /usr/sbin/sslwrap -nocert -addr
127.0.0.1 -port 901
#amanda dgram udp wait backup /usr/sbin/tcpd /usr/lib/amanda/amandad
#amandaidx stream tcp nowait backup /usr/sbin/tcpd /usr/lib/amanda/amindexd
#amidxtape stream tcp nowait backup /usr/sbin/tcpd /usr/lib/amanda/amidxtaped

# Esta linha ativa o tftpd "normal":
tftpd      dgram  udp    wait  nobody /usr/sbin/tcpd /usr/sbin/in.tftpd /tftpboot

# Esta linha ativa o tftpd-hpa, instalado por default no kts:
# tftpd dgram udp wait root /usr/sbin/in.tftpd -s /tftpboot -r blksize

bootps     dgram  udp    wait  root    /usr/sbin/bootpd    bootpd -i -t 120

```

Arquivo */opt/ltsp/i386/etc/ltsp.conf*

Este é o principal arquivo de configuração do LTSP, configure com atenção.
Substitua o "192.168.0.10" pelo endereço IP correto do servidor

```
[Default]
    SERVER          = 192.168.0.1
    XDM_SERVER      = 192.168.0.1
```

Esta opção faz com que o LTSP detecte automaticamente a placa de vídeo
dos terminais, utilizando os drivers do Xfree 4.2 ou X.org

```
XSERVER          = auto
```

Configure aqui o tipo de mouse que será usado nos terminais o default
é utilizar um mouse PS/2

```
X_MOUSE_PROTOCOL = "PS/2"
X_MOUSE_DEVICE   = "/dev/psaux"
X_MOUSE_RESOLUTION = 400
X_MOUSE_BUTTONS  = 3
USE_XFS          = N
LOCAL_APPS       = N
SCREEN_01        = startx
RUNLEVEL         = 5
```

#-----

Esta é a configuração individual dos terminais, que bipassam os valores
default determinados acima

A opção "Xserver = auto" utiliza a detecção automática do vídeo. Caso
ela não funcione em algum terminal, você pode indicar manualmente um driver,
substituir o "auto" por "vesa" por exemplo. O driver "vesa" costuma
funcionar na maioria das placas antigas, embora tenha um baixo desempenho.

O default é configurar as estações para utilizarem mouses PS/2
Exemplo para usar um mouse serial na estação:

```
#[ws001]
# XSERVER          = auto
# X_MOUSE_PROTOCOL = "Microsoft"
# X_MOUSE_DEVICE   = "/dev/ttyS0"
# X_MOUSE_RESOLUTION = 50
# X_MOUSE_BUTTONS  = 2
# X_MOUSE_EMULATE3BTN = Y
# USE_NFS_SWAP     = Y
```

```
# SWAPFILE_SIZE = 16m
# RUNLEVEL = 5
```

Exemplo para usar um mouse PS/2 COM RODA na estação:

```
#[ws001]
# XSERVER = auto
# X_MOUSE_PROTOCOL = "IMPS/2"
# X_MOUSE_DEVICE = "/dev/psaux"
# X_MOUSE_RESOLUTION = 400
# X_MOUSE_BUTTONS = 5
# X_ZAxisMapping = "4 5"
# USE_NFS_SWAP = Y
# SWAPFILE_SIZE = 32m
# RUNLEVEL = 5
```

Este exemplo força a estação a usar uma resolução e taxa
de atualização específicas, para o caso do X estar abrindo mas
o monitor estar ficando fora de sintonia (contribuição do ailtonjr):

```
#[ws001]
# XSERVER = auto
# X_MODE_0 = 800x600 #(Resolução)
# X_VERTREFRESH = 60 #(Refresh rate)
# X_COLOR_DEPTH = 16 #(Bits de Cor)
# USE_NFS_SWAP = Y
# SWAPFILE_SIZE = 32m
# RUNLEVEL = 5
```

Este é um exemplo de configuração para uma placa de vídeo Trident 9680,
usando o driver XF86_SVGA (do Xfree 3.3), que é o que oferece melhor
desempenho em micros muito antigos:

```
#[ws001]
# XSERVER = XF86_SVGA
# X_MOUSE_PROTOCOL = "Microsoft"
# X_MOUSE_DEVICE = "/dev/ttyS0"
# X_MOUSE_RESOLUTION = 50
# X_MOUSE_BUTTONS = 2
# X_MOUSE_EMULATE3BTN = Y
# X_MODE_0 = 800x600
# X_HORZSYNC = 30-54
# X_VERTREFRESH = 50-72
# X_COLOR_DEPTH = 16
# RUNLEVEL = 5
```

Existe ainda a opção de fazer funcionar uma placa de som espetada na estação.
Neste # caso qualquer som tocado no servidor,
sejam os sons dos programas ou um MP3 por exemplo serão transmitidos pela rede e

```
# tocados nas caixinhas da estação. Esta
# opção deve ser usada com cautela em redes com muitas estações pois os sons são
# transmitidos na forma de um wav descompactado
# através da rede. Cada estação cujo usuário esteja ouvindo um MP3 vai consumir
cerca # de 100 KB/s da banda da rede.
```

```
# As linhas abaixo ativam este recurso:
```

```
# [ws001]
#     SOUND                = Y
#     SOUND_DAEMON         = esd
#     VOLUME                = 80
#     MIC_VOLUME           = 80
#     CD_VOLUME            = 75
#     SMODULE_01           = auto
```

```
# Em placas onde a auto-deteção não funcione, você deve especificar manualmente os
# módulos que devem ser carregados
# para ativar o som na estação. Este é um exemplo, que ativa uma placa Sound Blaster
# ISA na estação:
```

```
#     SMODULE_01 = sound
#     SMODULE_02 = uart401
#     SMODULE_03 = sb io=0x220 irq=5 dma=1
```

```
# Este exemplo ativa o som onboard das placas baseadas no chipset nForce, na
# estação:
```

```
#     SMODULE_01 = sound
#     SMODULE_03 = i810_audio
```

```
# Ao ativar o compartilhamento do som, você deve configurar os programas para
# utilizarem o ESD como servidor de som ao invés
# do OSS, Alsa ou Arts. Procure esta opção nas preferências. Alguns programas, como
# o XMMS, Kaffeine e o GAIM suportam
# bem este recurso, enquanto outros não funcionam nem com reza brava, insistindo em
# tentar usar a placa de som do servidor.
# Você deve testar um a um os aplicativos que conseguem usar o som remotamente e
eliminar os demais.
```

```
# É possível também configurar o LTSP para permitir o uso de impressoras conectadas
# às estações. Neste caso, entra em ação o
# módulo "lp_server", que faz a estação desempenhar a função de um JetDirect, como
# se fosse um pequeno servidor de impressão,
# compartilhando a impressora conectada a ele com a rede.
# Para ativar o módulo de impressão, adicione as linhas abaixo na seção referente à
estação com a impressora:
```

```
# Para uma impressora paralela:
```

```
# [ws001]
# PRINTER_0_DEVICE = /dev/lp0
# PRINTER_0_TYPE = P
```

Para uma impressora USB (ao usar o Kernel 2.4 do LTSP na estação):

```
#[ws001]
# MODULE_01 = usb-ohci
# MODULE_02 = printer
# PRINTER_0_DEVICE = /dev/usb/lp0
# PRINTER_0_TYPE = S
```

Caso a estação esteja utilizando o Kernel da série 2.6, os módulos mudam de nome e a linha fica:

```
# [ws001]
# MODULE_01 = ohci-hcd
# MODULE_02 = usbip
# PRINTER_0_DEVICE = /dev/usb/lp0
# PRINTER_0_TYPE = S
```

A configuração no LTSP se resume a carregar os módulos de Kernel necessários para # ativar a impressora e indicar a porta a que ela está conectada. Isto faz com que a porta

da impressora fique disponível, para acesso a partir do servidor.

O próximo passo, é usar o Kaddprinterwizard, ou outro utilitário de configuração de

impressora, para instalar a impressora no servidor. Procure por uma opção

"Network Printer" (como no Kaddprinterwizard) ou "JetDirect Printer". Os

compartilhamentos do LTSP aparecem na rede exatamente da forma como a

impressora apareceria caso estivesse ligada a um JetDirect da HP,

usando inclusive a mesma porta, a 9100.

Este último exemplo ativa o compartilhamento de dispositivos locais para o disquete e

CD-ROM da estação. Ao usar o Debian, é criada uma pasta no desktop ("LTSP") com

os ícones para acessar os dispositivos em cada estação.

```
# [ws001]
# START_SAMBA = Y
# LOCAL_DEVICE_01 = /dev/hdc:cdrom
# LOCAL_DEVICE_02 = /dev/fd0:floppy
# RCFILE_01 = rc.localdev
# RUNLEVEL = 5
```

#-----

A configuração propriamente dita... :-)

Estes modelos abaixo mantém a configuração de utilizar um
mouse PS/2 (sem roda) usando a detecção automática do
driver de vídeo. Substitua a configuração pelos exemplos
acima conforme necessário.

Aqui estão as 6 estações usadas no projeto

As linhas comentadas são da parte de unidades locais

[ws001]

```
XSERVER                = auto
X_MOUSE_PROTOCOL       = "Microsoft"
X_MOUSE_DEVICE         = "/dev/ttyS0"
X_MOUSE_RESOLUTION    = 50
X_MOUSE_BUTTONS       = 2
X_MOUSE_EMULATE3BTN   = Y
X_MODE_0               = 800x600 #(Resolução)
X_VERTREFRESH         = 60 #(Refresh rate)
X_COLOR_DEPTH         = 24 #(Bits de Cor)
XkbModel               = ABNT2
XkbLayout              = br
# disquete e CD:
# START_SAMBA          = Y
# LOCAL_DEVICE_01      = /dev/hdc:cdrom
# LOCAL_DEVICE_02      = /dev/fd0:floppy
# RCFILE_01            = rc.localdev
# swap via rede:
# USE_NFS_SWAP         = Y
# SWAPFILE_SIZE        = 16m
# inicia em modo gráfico:
RUNLEVEL                = 5
```

[ws002]

```
XSERVER                = auto
X_MOUSE_PROTOCOL       = "Microsoft"
X_MOUSE_DEVICE         = "/dev/ttyS0"
X_MOUSE_RESOLUTION    = 50
X_MOUSE_BUTTONS       = 2
X_MOUSE_EMULATE3BTN   = Y
X_MODE_0               = 800x600 #(Resolução)
X_VERTREFRESH         = 60 #(Refresh rate)
X_COLOR_DEPTH         = 24 #(Bits de Cor)
XkbModel               = ABNT2
XkbLayout              = br
# disquete e CD:
# START_SAMBA          = Y
# LOCAL_DEVICE_01      = /dev/hdc:cdrom
# LOCAL_DEVICE_02      = /dev/fd0:floppy
# RCFILE_01            = rc.localdev
# swap via rede:
# USE_NFS_SWAP         = Y
# SWAPFILE_SIZE        = 16m
```

```
# inicia em modo gráfico:
RUNLEVEL = 5
```

[ws003]

```
XSERVER = XF86_SVGA
X_MOUSE_PROTOCOL = "Microsoft"
X_MOUSE_DEVICE = "/dev/ttyS0"
X_MOUSE_RESOLUTION = 50
X_MOUSE_BUTTONS = 2
X_MOUSE_EMULATE3BTN = Y
X_MODE_0 = 800x600 #(Resolução)
X_HORZSYNC = 30-54
X_VERTREFRESH = 50-72 #(Refresh rate)
X_COLOR_DEPTH = 16 #(Bits de Cor)
XkbModel = ABNT2
XkbLayout = br
# disquete e CD:
# START_SAMBA = Y
# LOCAL_DEVICE_01 = /dev/hdc:cdrom
# LOCAL_DEVICE_02 = /dev/fd0:floppy
# RCFILE_01 = rc.localdev
# swap via rede:
# USE_NFS_SWAP = Y
# SWAPFILE_SIZE = 16m
# inicia em modo gráfico:
RUNLEVEL = 5
```

[ws004]

```
XSERVER = auto
X_MOUSE_PROTOCOL = "Microsoft"
X_MOUSE_DEVICE = "/dev/ttyS0"
X_MOUSE_RESOLUTION = 50
X_MOUSE_BUTTONS = 2
X_MOUSE_EMULATE3BTN = Y
X_MODE_0 = 800x600 #(Resolução)
X_VERTREFRESH = 60 #(Refresh rate)
X_COLOR_DEPTH = 24 #(Bits de Cor)
XkbModel = ABNT2
XkbLayout = br
# disquete e CD:
# START_SAMBA = Y
# LOCAL_DEVICE_01 = /dev/hdc:cdrom
# LOCAL_DEVICE_02 = /dev/fd0:floppy
# RCFILE_01 = rc.localdev
# swap via rede:
# USE_NFS_SWAP = Y
# SWAPFILE_SIZE = 16m
# inicia em modo gráfico:
```

```

RUNLEVEL = 5

[ws005]
XSERVER = sis
X_MOUSE_PROTOCOL = "Microsoft"
X_MOUSE_DEVICE = "/dev/ttyS0"
X_MOUSE_RESOLUTION = 50
X_MOUSE_BUTTONS = 2
X_MOUSE_EMULATE3BTN = Y
X_MODE_0 = 800x600 #(Resolução)
X_VERTREFRESH = 60 #(Refresh rate)
X_COLOR_DEPTH = 24 #(Bits de Cor)
XkbModel = ABNT2
XkbLayout = br
# disquete e CD:
# START_SAMBA = Y
# LOCAL_DEVICE_01 = /dev/hdc:cdrom
# LOCAL_DEVICE_02 = /dev/fd0:floppy
# RCFILE_01 = rc.localdev
# swap via rede:
# USE_NFS_SWAP = Y
# SWAPFILE_SIZE = 16m
# inicia em modo gráfico:
RUNLEVEL = 5

```

```

[ws006]
XSERVER = cirrus
X_MOUSE_PROTOCOL = "Microsoft"
X_MOUSE_DEVICE = "/dev/ttyS0"
X_MOUSE_RESOLUTION = 50
X_MOUSE_BUTTONS = 2
X_MOUSE_EMULATE3BTN = Y
X_MODE_0 = 800x600 #(Resolução)
X_VERTREFRESH = 60 #(Refresh rate)
X_COLOR_DEPTH = 24 #(Bits de Cor)
XkbModel = ABNT2
XkbLayout = br
# disquete e CD:
# START_SAMBA = Y
# LOCAL_DEVICE_01 = /dev/hdc:cdrom
# LOCAL_DEVICE_02 = /dev/fd0:floppy
# RCFILE_01 = rc.localdev
# swap via rede:
# USE_NFS_SWAP = Y
# SWAPFILE_SIZE = 16m
# inicia em modo gráfico:
RUNLEVEL = 5

```

```

#-----
# EXEMPLO de configuração para uma estação usando o driver
# "vesa", mouse serial e resolução de 800x600 com 16 bits
# de cor, 60 hz de refresh para o monitor e ainda ativar
# a placa de som:

# [ws001]
# XSERVER = vesa
# X_MODE_0 = 800x600 #(Resolução)
# X_VERTREFRESH = 60 #(Refresh rate)
# X_COLOR_DEPTH = 16 #(Bits de Cor)
# X_MOUSE_PROTOCOL = "Microsoft"
# X_MOUSE_DEVICE = "/dev/ttyS0"
# X_MOUSE_RESOLUTION = 50
# X_MOUSE_BUTTONS = 2
# X_MOUSE_EMULATE3BTN = Y
# USE_NFS_SWAP = Y
# SWAPFILE_SIZE = 32m
# SOUND = Y
# SOUND_DAEMON = nasd
# VOLUME = 80
# MIC_VOLUME = 80
# CD_VOLUME = 75
# SMODULE_01 = soundcore
# SMODULE_02 = uart401
# SMODULE_03 = sb io=0x220 irq=5 dma=1
# RUNLEVEL = 5

```

```

# EXEMPLO de configuração para uma estação usando a detecção
# automática do vídeo, mouse PS/2 com roda e resolução de
# 1024x768 com 16 bits de cor e 60 hz de refresh para o monitor:

```

```

# [ws002]
# XSERVER = auto
# X_MODE_0 = 1024x768 #(Resolução)
# X_VERTREFRESH = 60 #(Refresh rate)
# X_COLOR_DEPTH = 16 #(Bits de Cor)
# X_MOUSE_PROTOCOL = "IMPS/2"
# X_MOUSE_DEVICE = "/dev/psaux"
# X_MOUSE_RESOLUTION = 400
# X_MOUSE_BUTTONS = 5
# X_ZAxisMapping = "4 5"
# USE_NFS_SWAP = Y
# SWAPFILE_SIZE = 32m
# RUNLEVEL = 5

```

```

# Swap via rede.
# As opções:
# USE_NFS_SWAP = Y

```

```
# SWAPFILE_SIZE = 32m
```

```
# Fazem com que seja criado um arquivo swap no HD do servidor,  
# que fica disponível para a estação. Este arquivo é recomendável  
# em estações com menos de 32 MB de memória RAM e é utilizado  
# apenas nos momentos em que a estação fica sem memória disponível.  
# Você pode alterar o número para o valor desejado ou desativar a  
# opção alterando a primeira linha para:
```

```
# USE_NFS_SWAP = N
```