

**UNISA – UNIVERSIDADE DE SANTO AMARO  
SISTEMAS DE INFORMAÇÃO**

**Daniela Cristina Bastos  
Francisco Erlânio de Lima  
Paulo Eduardo Chiva de Lima  
Samuel Braz de Sousa**

**O USO DE THIN CLIENTS COMO FERRAMENTA  
DE BAIXO CUSTO PARA INCLUSÃO DIGITAL**

**São Paulo**

**2012**



**Daniela Cristina Bastos**  
**Francisco Erlânio de Lima**  
**Paulo Eduardo Chiva de Lima**  
**Samuel Braz de Sousa**

**O USO DE THIN CLIENTS COMO FERRAMENTA  
DE BAIXO CUSTO PARA INCLUSÃO DIGITAL**

Trabalho de Conclusão de Curso apresentado para a  
obtenção do título de Bacharel em Sistemas da  
Informação da Universidade de Santo Amaro, sob  
orientação do Prof. Ms. Marlom Alves Konrath.

**São Paulo**  
**2012**

**Daniela Cristina Bastos**  
**Francisco Erlânio de Lima**  
**Paulo Eduardo Chiva de Lima**  
**Samuel Braz de Sousa**

**O USO DE THIN CLIENTS COMO FERRAMENTA  
DE BAIXO CUSTO PARA INCLUSÃO DIGITAL**

Trabalho de Conclusão de Curso apresentado para a obtenção do título de Bacharel em Sistemas da Informação da Universidade de Santo Amaro, sob orientação do Prof. Ms. Marlom Alves Konrath.

Data de aprovação \_\_\_\_/\_\_\_\_/\_\_\_\_

**BANCA EXAMINADORA**

---

Prof.<sup>a</sup> Ms. Denise Cordeiro Gonçalves Canal

---

Prof. Ms. Itsche Baran

---

Prof. Ms. Marlom Alves Konrath

CONCEITO FINAL: \_\_\_\_\_

## AGRADECIMENTOS

Agradecemos a todos os professores que no decorrer do curso nos proporcionaram o conhecimento necessário para que pudéssemos produzir este trabalho, em especial ao professor Marlom Konrath, que com seu profissionalismo, ética e respeito conseguiu fazer a diferença no curso. Agradecemos, também, aos nossos colegas de classe, em especial ao nosso amigo Luiz Anacleto, que, inclusive, nos forneceu peças e equipamentos que foram essenciais para a implementação da rede utilizada.

Daniela - Acima de tudo agradeço a Deus por me dar a oportunidade de exercer essa profissão magnífica. À minha mãe e minha irmã, que sempre acreditaram muito no meu trabalho e me ajudaram no que foi preciso. Aos meus amigos Francisco, Paulo e Samuel que estiveram comigo nessa batalha durante esses longos quatro anos. Agradeço a todos os professores, amigos e colegas que de alguma maneira contribuíram para essa realização.

Francisco - Agradeço primeiramente a Deus, por ter me proporcionado essa oportunidade, e a todas as pessoas que fizeram e fazem parte desta vitoriosa conquista, principalmente aos colegas de classe, que nesses quatro anos se tornaram como irmãos, em especial ao amigo Paulo Eduardo Chiva, que com toda a sua paciência sempre me ajudou nos momentos que precisei. Agradeço em especial à minha esposa Viviane Domingos, que contribuiu muito para que esse momento especial acontecesse.

Paulo – Primeiramente, agradeço à minha família, a qual me deu a base necessária para chegar até aqui. Agradeço ao carinho e paciência de meu filho Eduardo Santos Chiva, o qual muitas vezes, devido à realização deste trabalho, se privou da presença do pai. Agradeço em especial à minha “esposa”, companheira e amiga Marcia Brady, que com amor e carinho únicos fez o possível e o impossível para que eu tivesse o máximo de tempo disponível para me dedicar a este trabalho.

Samuel - Agradeço a Deus que é o autor de tudo e de todos, pois sem Ele nada seríamos a todos os alunos que fizeram parte desta turma aos companheiros deste grupo que com o esforço e dedicação de cada um conseguimos chegar ao nosso objetivo. Quero agradecer em especial a minha querida Esposa Léia Ribeiro que com toda paciência me apoiou nesses quatro anos de estudos aos meus amados filhos pelo qual tenho feito todo esforço para educá-los.

*“Aos esfarrapados do mundo e aos que neles se descobrem e, assim descobrindo-se, com eles sofrem, mas, sobretudo, com eles lutam.”*

Paulo Freire

## RESUMO

As pessoas, no Brasil e no mundo, estão utilizando cada vez mais computadores e suas redes para se comunicarem e buscar informações. Mesmo no mercado de trabalho, a maioria das áreas acaba por ter a necessidade de utilização de computadores para a automação de tarefas, o que acaba por exigir conhecimentos de informática. Há também um terço da população considerada excluída digitalmente, o que implica em uma parcela significativa de pessoas com poucas chances de se inserir no mercado de trabalho, resultando em mais problemas sociais. Em contrapartida, a demanda dos softwares produzidos por hardwares cada vez mais potentes resulta na obsolescência cada vez mais prematura de computadores, que muitas vezes são descartados quando, com o uso correto poderiam funcionar como excelentes ferramentas de inclusão digital. Partindo destas questões, este trabalho tem como objetivo demonstrar a viabilidade de criação de uma rede de computadores, modelo cliente-servidor, com a utilização de *thin clients*, de modo a reaproveitar computadores considerados obsoletos, criando, em conjunto com softwares livres, uma solução de baixo custo para o combate da exclusão digital.

**Palavras-chave:** inclusão digital, DRBL, LTSP, rede de computadores, *thin client*.

## ABSTRACT

People in Brazil and all around the world are increasingly using their computers and networks to communicate and search information. Even in the market, most areas require the usage of computers to automate tasks, which leads to the need of computer skills. Besides that, a third of the population is considered digitally excluded, what implies in a significant portion of people with few chances of entering the job market, resulting in increasing social problems. In contrast, the constant software demand for more powerful hardware results in the even more premature obsolescence of computers, that are often discarded when, with proper use, could serve as an excellent tool for digital inclusion. Based on these questions, this work aims to demonstrate the feasibility of creating a network of computers, in a client-server model, which uses the thin client model, as a way to reuse computers considered as obsolete, creating, with the support of free software, a low cost solution to fight digital exclusion.

**Keywords:** digital inclusion, DRBL, LTSP, computer networks, thin client.

## LISTA DE FIGURAS

Figura 1 – O Modelo de referência OSI.....	18
Figura 2 - Thin client da Intel.....	23
Figura 3 - Funcionamento do XDMCP.....	26
Figura 4 - Ordem dos cabeçalhos .....	28
Figura 5 - Pacote RRQ/WRQ .....	28
Figura 6 - Pacote DATA .....	28
Figura 7 - Pacote ACK.....	29
Figura 8 - Pacote ERROR.....	29
Figura 9 - Funcionamento do DHCP .....	31
Figura 10 - Interceptação de dados em uma rede.....	34
Figura 11 - Envio criptografado dos dados usando o ssh .....	35
Figura 12 - Resultado do comando ltsp-info.....	37
Figura 13 - BIOS de um <i>netbook</i> ajustado para realizar boot pela rede.....	38
Figura 14 - Placa de Rede com <i>slot</i> e EPROM .....	39
Figura 15 - Tela de login do LTSP exibida no thin client .....	40
Figura 16 - Instalação do Gimp no Debian: listagem de pacotes dependentes.....	43
Figura 17 - Ciclo de lançamento e suporte do Ubuntu .....	44
Figura 18 - <i>Switch</i> de oito portas utilizado nos testes.....	45
Figura 19 - Instalação do DRBL sem a chave de assinatura.....	51
Figura 20 - Ajuste da BIOS para <i>boot</i> via PXE.....	71
Figura 21 - Gerador de imagens do site <a href="http://www.rom-o-matic.net">www.rom-o-matic.net</a> .....	72
Figura 22 - Ajuste da BIOS para boot via CDROM .....	73
Figura 23 - Exibição da tela de <i>login</i> das estações <i>thin client</i> .....	73
Figura 24 - Mapa da rede.....	75

## LISTA DE GRÁFICOS E TABELAS

Gráfico 1 - Evolução do consumo de memória durante as Etapas de 1 a 4 .....	63
Gráfico 2 - Consumo de memória com carregamento sequencial das estações TC01 e TC02 .....	64
Gráfico 3 - Uso do processador no servidor LTSP .....	66
Gráfico 4 - Uso do processador no servidor DRBL .....	66
Gráfico 5 - Tráfego de dados .....	67
Tabela 1 - Motivos pelo não-uso de computadores e Internet .....	12
Tabela 2 - Internet no Brasil: locais de uso .....	13
Tabela 3 - Lista de eventos do TFTP .....	27
Tabela 4 - Especificações do Servidor e thin clients .....	46
Tabela 5 - Memória adicional necessária ao servidor .....	65
Tabela 6 - Requerimentos de hardware para instalação da versão Desktop do Ubuntu 12.04 .....	69
Tabela 7 - Configurações do Servidor LTSP .....	70
Tabela 8 - Configuração de hardware dos <i>thin clients</i> .....	74

# SUMÁRIO

1	INTRODUÇÃO .....	11
1.1	Motivação.....	12
1.2	Objetivos .....	14
1.3	Estrutura do Trabalho.....	15
2	REVISÃO BIBLIOGRÁFICA.....	16
2.1	Redes de Computadores .....	16
2.1.1	Modelo de Referencia OSI.....	17
2.1.1.1	Camada Física .....	18
2.1.1.2	Camada de Enlace de Dados .....	19
2.1.1.3	Camada de Rede .....	19
2.1.1.4	Camada de transporte.....	19
2.1.1.5	Camada de Sessão.....	19
2.1.1.6	Camada de Apresentação.....	20
2.1.1.7	Camada de Aplicação .....	20
2.1.2	Modelo TCP/IP .....	20
2.2	Thin Client <i>versus</i> Thick Client.....	22
2.2.1	Thin Client.....	22
2.2.2	Thick Client.....	24
2.2.3	Principais protocolos utilizados pelas soluções <i>thin client</i> .....	25
2.2.3.1	XDMCP .....	25
2.2.3.2	TFTP .....	26
2.2.3.3	PXE .....	29
2.2.3.4	DHCP .....	30
2.2.3.5	NBD .....	32
2.2.3.6	SSH.....	33
2.3	Soluções <i>thin client</i> .....	35
2.3.1	LTSP .....	35
2.3.1.1	Funcionamento do LTSP.....	38
2.3.2	DRBL.....	40
3	ESTUDO DE CASO .....	42
3.1	Sistema operacional utilizado.....	42
3.2	Equipamentos utilizados para o estudo de caso .....	45

3.3	Instalando e Configurando o LTSP .....	46
3.4	Instalando o DRBL.....	50
3.5	Análise de desempenho das soluções propostas.....	61
3.5.1	Análise do consumo de memória .....	61
3.5.2	Análise do consumo de processamento.....	62
3.5.3	Análise do tráfego de rede .....	62
3.6	Análise dos dados coletados .....	63
3.6.1	Memória .....	63
3.6.2	Processador.....	65
3.6.3	Tráfego de rede.....	67
3.7	Solução escolhida.....	68
4	RESULTADOS .....	70
4.1	O ambiente utilizado .....	70
4.2	Configurações Iniciais.....	71
4.3	Implementação .....	74
4.4	Validação .....	77
4.5	Conclusões das Validações .....	78
5	CONCLUSÃO E TRABALHOS FUTUROS.....	80
5.1	Trabalhos futuros.....	81
	REFERÊNCIAS BIBLIOGRÁFICAS.....	83
	ANEXOS.....	87

# 1 INTRODUÇÃO

Atualmente a questão da inclusão digital é algo discutido mundialmente, fazendo, inclusive, parte das Metas do Milênio estabelecidas pela ONU, onde metas de conectividade, através da internet, aos cidadãos de cada país, foram estabelecidas.

De acordo com pesquisa realizada pela Fundação Getúlio Vargas (CENTRO DE POLÍTICAS SOCIAIS, 2012), atualmente o Brasil encontra-se exatamente na média mundial, com apenas 33% da população com acesso à Internet, embora, mesmo internamente, haja uma grande discrepância entre as regiões, com cidades onde 74% da população possuem computador com acesso à rede mundial de computadores em seus domicílios, e cidades na qual o percentual de acesso é praticamente nulo.

Existem muitas iniciativas para a erradicação da exclusão digital, sejam elas realizadas pelos governos, como os telecentros da Prefeitura de São Paulo (SÃO PAULO, 2012), sejam pela iniciativa privada, como o programa Conecta, da operadora de celulares Oi/TNPL PS S/A (OI CONECTA, 2012). No entanto, a dimensão geográfica e a falta de recursos apropriados fazem com que essas iniciativas deixem de atingir adequadamente grande parte da população.

Com a evolução dos computadores, a sociedade passou por uma profunda transformação. Houve uma mudança na forma com que as pessoas trabalham, interagem e se comunicam. Para pessoas mais jovens, chega a ser impensável o funcionamento de grandes corporações, como, por exemplo, instituições bancárias, sem a utilização de computadores para organização das diversas informações necessárias ao seu processo de funcionamento. Da mesma forma, diversos órgãos governamentais evoluíram ao ponto de prestar diversos de seus serviços também pela Internet, como por exemplo, o agendamento para retirada do Título Eleitoral realizado pela Justiça Eleitoral. Já outros possuem serviços que só podem ser realizados através do uso de computador e Internet, como a Receita Federal, onde a entrega da declaração de imposto de renda passou, a partir de 2011, a ser aceita apenas por meio eletrônico, devido à pequena quantidade de pessoas que realizavam a entrega através de outras formas.

## 1.1 Motivação

Como as mudanças da sociedade advindas do uso da Internet e popularização dos computadores em um curto prazo de tempo, estas mudanças acabaram por gerar uma grande quantidade de pessoas excluídas tecnologicamente. De acordo com CENTRO DE POLÍTICAS SOCIAIS (2012), em uma pesquisa realizada em todo o território nacional com dados obtidos a partir do CENSO de 2010, a somatória de pessoas consideradas excluídas tecnologicamente que declararam o motivo como “não tinha acesso ao computador”, “o custo de um microcomputador era alto” e “o custo de utilização da Internet era alto”, atinge a somatória de 31,95%, como pode ser visto na Tabela 1.

Tabela 1 - Motivos pelo não-uso de computadores e Internet

<b>Razões da exclusão digital</b>		
<b>Motivo</b>	<b>Total</b>	
Não achava necessário ou não quis	33.876.749	33,14%
Não sabia utilizar a Internet	32.150.424	31,45%
Não tinha acesso a microcomputador	30.454.340	29,79%
O custo de um microcomputador era alto	1.799.587	1,76%
Outro motivo	1.556.277	1,52%
O custo de utilização da Internet era alto	411.217	0,40%
O microcomputador que usa em outro local não estava conectado à Internet	321.758	0,31%
<b>Total:</b>	<b>102.232.483</b>	<b>100%</b>

Fonte: CENTRO DE POLÍTICAS SOCIAIS, 2012

Este valor aproxima-se de um terço de toda a população do país. Ou seja, ao se observar apenas no que tange às exigências do mercado de trabalho, que busca pessoas capacitadas para uso de computador e Internet, sejam em pequenas, médias ou grandes empresas, temos um terço de toda a população já automaticamente consideradas inaptas para a obtenção de vagas de trabalho nestas áreas.

Outro ponto a ser observado é que, da parcela da população que utiliza a Internet, conforme pode ser visto na Tabela 2, 41,84% realizam o acesso através de

outros locais que não suas residências, como centros públicos de acesso, sejam eles pagos ou gratuitos, casa de amigos e parentes ou em estabelecimento em que frequentava algum curso. Isso mostra a equivalência do que pode ser chamado “computador coletivo” em contrapartida ao computador pessoal, e evidencia que os centros públicos de acesso são uma excelente ferramenta de combate à exclusão digital.

Tabela 2 - Internet no Brasil: locais de uso

<b>Locais de Uso</b>	
<b>Nos últimos três meses, utilizou internet em:</b>	
seu domicílio por banda larga	<b>46,92%</b>
centro público de acesso pago	<b>35,11%</b>
seu trabalho	<b>31,03%</b>
domicílio de parentes, amigos ou outro local	<b>19,71%</b>
estabelecimento em que freqüentava algum curso	<b>17,50%</b>
seu domicílio por acesso discado	<b>11,24%</b>
centro público de acesso gratuito	<b>5,52%</b>

Fonte: CENTRO DE POLÍTICAS SOCIAIS, 2012.

Paralelamente a isso, há também a questão da obsolescência do hardware. Segundo Davis (2008), PCs e Notebooks têm seu ciclo de uso encerrado em três ou quatro anos. A demanda por atualizações de softwares, seja por questões comerciais, seja por questões de segurança, resultam na troca do hardware pelas pessoas ou empresas, faz com que um equipamento funcional seja, em um curto espaço de tempo, considerado obsoleto. No entanto, estes equipamentos não estão inoperantes e seu uso ainda pode (e deve) ser considerado. Na maioria das vezes estes computadores acabam descartados, mas, acredita-se, poderiam ser reutilizados para criação de centros públicos de acesso à Internet, desde que utilizando softwares que possam aproveitar de forma funcional o seu uso, criando-se, assim, uma ferramenta de baixo custo para a inclusão digital.

## 1.2 Objetivos

Este trabalho apresenta o resultado de um estudo sobre a viabilidade de montagem de um laboratório de informática de baixíssimo custo, tanto de software como hardware, o qual irá utilizar como estações computadores considerados obsoletos por empresas e/ou indivíduos. Como resultado deste trabalho foi montado um laboratório de informática para aplicação dos resultados obtidos.

Além de apresentar o processo que culminou na elaboração do laboratório, este trabalho buscou ainda, como objetivos específicos, os itens apresentados a seguir:

- Desenvolver os conceitos sobre o funcionamento de redes, bem como da solução *thin client*;
- Analisar duas soluções de *thin clients* não-comerciais, de modo a buscar qual melhor se adequava para a criação do laboratório de informática pretendido. Para isso, foram analisados três fatores: o consumo de CPU, consumo de memória e tráfego de rede;
- Apontar as vantagens e desvantagens de uma solução *thin client* em relação a uma solução *thick client*;
- Mostrar as etapas e ferramentas necessárias para a implementação de uma solução baseada em *thin client*.

Existem trabalhos semelhantes, como FARIA (2009), no qual foi efetuada a comparação entre soluções, ou como BELLO (2008), que apresentou um modelo de rede para uso em uma escola, porém, em ambos não há um detalhamento adequado do funcionamento dos *thin clients*, não havendo um estudo mais aprofundado as necessidades de recursos de memória, processamento ou tráfego de rede, além de utilizarem versões consideradas defasadas dos programas necessários, não sendo possível executá-las com a utilização das versões mais recentes dos softwares.

O custo de implantação da rede é próximo de zero, já que as estações são todas resultantes de computadores doados, já não mais utilizados por serem considerados obsoletos, além da utilização de software livre, não havendo custo com licenças.

Os *thin clients*, como ferramenta de inclusão digital, podem tanto instalar uma infra-estrutura necessária para inclusão digital em qualquer posto público que se proponha a aplicá-la, como para maximizar a infra-estrutura que porventura possa já existir. Além disso, uma das principais vantagens do uso de *thin clientes* é a possibilidade do reaproveitamento de computadores mais antigos como estações de trabalho, mitigando assim parte do um ciclo de obsolescência programada de hardware. Com estes dois fatores, consegue-se proporcionar à população excluída digitalmente, um mínimo de proficiência na utilização do computador e a Internet.

### **1.3 Estrutura do Trabalho**

Este trabalho está organizado em cinco capítulos, além da referência bibliográfica e os anexos. Além deste capítulo, temos o capítulo 2, onde são mostrados o funcionamento das redes de computadores e seus diversos protocolos (2.1), a diferença entre as redes comuns com relação às redes *thin client* (2.2) e as principais soluções *thin client* livres existentes no mercado, como o LTSP e o DRBL, além de um comparativo entre ambos. A seguir, o capítulo 3 apresenta um estudo de caso, mostrando o sistema operacional escolhido (3.1), a instalação das soluções LTSP (3.3) e DRBL (3.4) e os equipamentos necessários para tanto (3.2), bem como a análise de desempenho das soluções propostas e dos resultados (3.5 e 3.6). No capítulo 4 são apresentados os resultados, a partir da implementação da solução (4.3), a validação necessária (4.4) e as conclusões da mesma (4.5). Por último, o capítulo 5 apresenta a conclusão, bem como trabalhos futuros (5.1).

## 2 REVISÃO BIBLIOGRÁFICA

O capítulo 2 trata da revisão bibliográfica, abordando as redes de computadores (seção 2.1), apresentando sua definição e explicando o seu funcionamento. O capítulo abordará também o TCP/IP (seção 2.1.3), além da comparação entre os *thin client* e os *thick clients* (seções 2.2 e 2.2.2), abordando suas vantagens, desvantagens, principais protocolos (seção 2.2.3) e soluções duas soluções de thin client, na qual é o objetivo de deste trabalho, apresentando e explicando o funcionamento do LTSP (seção 2.3.1.1) e do DRBL.

### 2.1 Redes de Computadores

As redes de computadores predominam, nos dias de hoje, não só em empresas, mas também em residências, principalmente com o objetivo de compartilhamento de recursos, como impressoras e *Internet*.

Segundo Tanenbaum (2003) uma rede de computadores é “um conjunto de computadores autônomos interconectados por uma única tecnologia”. Ou seja, o termo redes de computadores é descrito como o meio na qual os computadores são conectados uns aos outros e a forma como trocam informações podendo assim, fazer com que seus usuários compartilhem arquivos, programas, jogos e periféricos tais como impressora etc.

Dentro deste processo é preciso que todos os computadores falem a mesma língua. Para que isso ocorra é necessário uma metodologia e protocolos de comunicação de redes sem os quais não é possível realizar uma conexão de rede. Os protocolos de rede, por exemplo, são responsáveis em transmitir os pacotes de dados e que eles cheguem ao seu destino final. Sendo assim todos os dispositivos que estão ligados à rede precisam ter os mesmos protocolos instalados para que ocorra uma comunicação eficaz.

### 2.1.1 Modelo de Referencia OSI

O Modelo OSI, apresentado na Figura 1, é formado por sete camadas que são usadas para permitir e facilitar a transmissão de dados em uma rede (TANENBAUM, 2003). Esta divisão segue um padrão internacional, estabelecido pela ISO. Para uma melhor compreensão alguns princípios foram aplicados para se chegar a essas camadas. São eles:

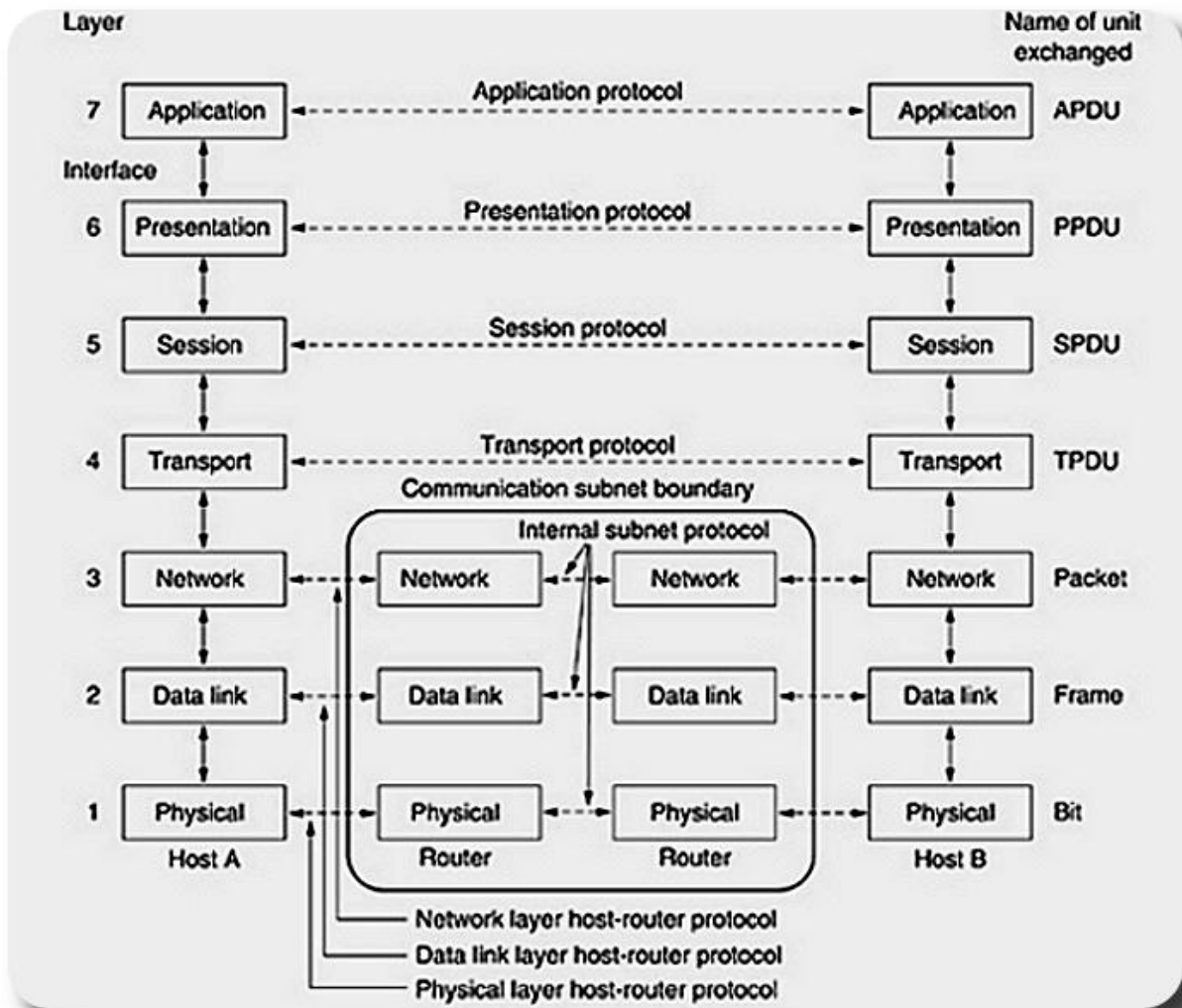
1. Uma camada deve ser criada onde houver necessidade de outro grau de abstração.
2. Cada camada deve executar uma função bem definida.
3. A função de cada camada deve ser escolhida tendo em vista a definição de protocolos padronizados internacionalmente.
4. Os limites da camada devem ser escolhidos para reduzir o fluxo de informações transportadas entre as interfaces.
5. O número de camadas deve ser suficientemente grande para que funções distintas não precisem ser desnecessariamente colocadas na mesma camada e suficientemente pequeno para que a arquitetura não se torne difícil de controlar.

(Tanenbaum, 2003, p.41)

As camadas do modelo OSI são assim divididas:

- Camada física;
- Camada de Enlace de Dados;
- Camada de Rede;
- Camada de Transporte;
- Camada de Sessão;
- Camada de apresentação;
- Camada de Aplicação.

Figura 1 – O Modelo de referência OSI



Fonte: (Tanenbaum, 2004, p.41)

A seguir, cada uma destas camadas é brevemente abordada.

### 2.1.1.1 Camada Física

É responsável pela transmissão e pela recepção de bits brutos não estruturados através de um meio físico. Bit representa a sigla para *Binary Digit*, em português, dígito binário. Largamente usado na computação; é a menor unidade de informação que pode ser armazenada ou transmitida. Um bit pode assumir apenas um de dois valores possíveis: 0 ou 1.

Dessa forma a Camada Física recebe os bits e garante que na rede as outras camadas recebam o bit com o mesmo valor facilitando assim a transmissão de

dados.

### **2.1.1.2 Camada de Enlace de Dados**

Esta camada tem como tarefa fazer uma comunicação direta entre duas interfaces na mesma rede e transmitir o pacote de dados de forma correta. Para isso é enviado ao receptor um pacote de dados em quadros sequenciais e se estiver correto o receptor envia um pacote de dados de confirmação, sendo assim após a confirmação os dados são enviados de forma correta.

### **2.1.1.3 Camada de Rede**

Tem como responsabilidade controlar e decidir que caminho físico os dados devem ser levados tendo como base as condições de rede e as prioridades do serviço.

### **2.1.1.4 Camada de transporte**

Recebe os dados da camada acima dela; se necessário divide esses dados em pedaços menores e envia-os para a camada de rede garantindo que esses dados chegarão e serão remontados corretamente na outra extremidade.

Esta camada é também chamada de camada fim a fim, ou seja, ela liga a origem ao destino para que o usuário envie e receba seus dados com confiabilidade.

### **2.1.1.5 Camada de Sessão**

Esta camada permite que os usuários de uma rede consigam estabelecer

sessões entre eles através de uma conexão de transporte, também controla a troca de dados e sincroniza as operações para que numa possível falha de transmissão não se perca os dados.

#### **2.1.1.6 Camada de Apresentação**

A Camada de Apresentação está relacionada com a sintaxe dos dados transmitidos, pois esse serviço gerencia a entrada, a troca e controle dos dados estruturados. Esta camada resolve o problema de diferença de sintaxe entre os sistemas

#### **2.1.1.7 Camada de Aplicação**

A Camada de Aplicação possui uma série de protocolos necessários para que o usuário consiga utilizar a rede, um dos mais utilizados é o HTTP, acrônimo de *HyperText Transfer Protocol*, considerado como o protocolo base para se navegar na web.

A transmissão de dados utilizando Modelo OSI, de uma maneira mais simples, é feita da seguinte maneira: cada camada tem a função de adicionar um cabeçalho aos dados do usuário que está transmitindo desta maneira a função das outras camadas é retirar os cabeçalhos dos dados que chegam e entregá-los ao nível superior em sua forma original, agilizando assim o processo de transmissão.

#### **2.1.2 Modelo TCP/IP**

Definido pela primeira vez em 1974, o modelo TCP/IP (TANENBAUM, 2003) foi usado pela rede do Departamento de Defesa dos Estados Unidos com a preocupação de detectar a perda de pacotes que trafegavam pela rede e fornecer

uma arquitetura flexível para a transferência de dados.

O modelo TCP/IP na realidade, é um conjunto de protocolos, entre eles, como o próprio nome diz, o TCP (*Transmission Control Protocol* – Protocolo de Controle de Transmissão) e o IP (*Internet Protocol* – Protocolo Internet). É atualmente o padrão *de facto* mundial para comunicação de sistemas abertos.

Segundo Tanenbaum (2003), o TCP/IP está organizado em quatro camadas, a saber:

- **Aplicação:** é a camada onde os programas “se comunicam”. Diversos protocolos são utilizados nesta camada, como o SMTP (usado para envio de e-mail) e o DNS (usado para conversão de nomes de domínio em endereços IP e vice-versa), entre outros. Cada tipo de programa se comunica com um protocolo diferente dentro desta camada. Esta camada se comunica com a camada de Transporte.
- **Transporte:** Esta camada recebe os dados da camada de Aplicação e os divide em pedaços de tamanho pequeno (chamados pacotes). A seguir, coloca-os em ordem e os envia para a próxima camada (Internet). É responsável também pelo processo inverso, ou seja, receber os pacotes da camada Internet, remonta-los e os entregar para a camada de Aplicação.
- **Internet:** Nesta camada é onde fica o protocolo IP. Ela recebe os pacotes da camada superior e adiciona o endereço do computador que esta enviando e o endereço que receberá os dados. Esses endereços virtuais são chamados de endereço de IP. Após a adição dos cabeçalhos, o pacote é então enviado à camada de Interface com Rede.
- **Interface com Rede:** A função desta camada é enviar os dados recebidos para a rede, preocupando-se com as características e limitações do meio físico sendo utilizado. Atualmente a maioria dos computadores utiliza a rede Ethernet, mas outras tecnologias como as redes sem fio estão se popularizando rapidamente.

## 2.2 Thin Client versus Tick Client

Nesta seção, serão apresentadas as definições, características, vantagens e desvantagens das redes baseadas em *thin clients*, objeto deste trabalho, e as tradicionais redes *tick clients*.

### 2.2.1 Thin Client

A tecnologia utilizada pelo *Thin Client* é uma solução que busca aperfeiçoar o funcionamento de um computador. . É projetado para ser utilizado em cenários em que se tem o uso de computadores com recursos de hardware limitados. Nestes a execução das suas aplicações é realizada de forma remota, em um servidor, transmitindo-se a tela com os resultados ao cliente, através de um protocolo leve. O thin client pode ser definido como:

Um *Thin Client* nada mais é do que um PC desprovido de CD-ROM, leitor de disquete, placas adicionais ou disco rígido. Estes dispositivos são gerenciados de forma centralizada. A partir do momento em que todos os aplicativos estão rodando de um servidor central, e não do thin client, este não requer recursos mais poderosos para si mesmos. (Ansari, Tiwari e Agrawal, 2005, p. 2)

Portanto, o cenário ideal de uso do *thin client* é aquele em que o usuário possui um hardware obsoleto, inferior ao de uma máquina normal. O *thin client* age então como um terminal burro, totalmente dependente dos recursos de outro computador (servidor), para seu funcionamento. É necessário que seja criado uma imagem de boot, para que assim se possa iniciar a comunicação entre as máquinas que estão ligadas na rede. Esse boot geralmente é rápido e programado pelo administrador da rede, durando apenas segundos até a abertura das aplicações. Um ponto considerado bastante positivo no uso dessa solução é o fato de não haver armazenamento em disco rígidos, dessa maneira impossibilita que algo seja corrompido por software malicioso (também conhecido como Vírus ou Malware), ou até mesmo por alguma incompatibilidade. Outro fato bastante positivo é poder restringir o acesso dos usuários, fazendo com que eles acessem apenas aplicativos

que foram previamente autorizados pelo administrador do sistema. O computador usado como servidor, é um computador central responsável pelo processamento das aplicações e armazenamento de arquivos. Já os computadores utilizados pelos usuários, são computadores que normalmente possuem poucos recursos.

As soluções *thin client* focam na segurança e economia, pois em caso de problemas em alguma das estações de trabalho, basta substituí-las, não havendo necessidade de se instalar ou configurar aplicativos.

Praticamente qualquer computador capaz de acessar a rede pode ser utilizado como thin client. Há ainda soluções de baixo custo customizadas, que utilizam hardware barato e são montadas e disponibilizadas como um pequeno dispositivo ao qual se conectam, teclado, mouse e monitor. Um exemplo deste último caso é apresentado na Figura 2.

Figura 2 - Thin client da Intel



Fonte: THIN CLIENT, 2012

Conforme Greenberg (2000) a IBM foi uma das primeiras empresas a batizarem a nova solução, chamado-o de “cliente magro”: o *Thin Client*.

Segundo Greenberg (2000), as principais características e vantagens dos *thin clients* são:

- Custo minimizado com manutenção.
- Facilidade de gerenciamento dos dados.
- Permite simplificar todo o processo de instalação e gestão de clientes.

- Segurança de dados e aplicações.
- Controle de acesso a software.
- Baixo consumo elétrico.
- Baixo custo com licença de software.
- Produz menos ruído do que um computador convencional.
- Não há necessidade de ser substituído com a mesma frequência de um computador normal, diminuindo o aumento de lixo eletrônico.

Já as principais desvantagens desta solução são:

- Caso haja algum problema com o servidor, os thin clients não funcionarão.
- Limitação no uso de portas USB.

Em suma, um *thin client* é uma solução que utiliza um computador em rede como servidor e vários computadores como cliente. É uma solução projetada para ter baixo custo sendo os aplicativos totalmente processados no servidor. As estações *thin client*, são preparadas com um software cliente, monitor, teclado e mouse e a comunicação é feita via uma rede de computadores. É importante ressaltar, que os usuários utilizam as estações de trabalho de forma individual, ou seja, o fato de estarem em rede e dependendo do servidor para seu funcionamento não obriga ou impede que cada usuário trabalhe de forma independente.

### 2.2.2 Thick Client

O *Thick Client*, diferentemente do *Thin Client*, tem seu próprio sistema operacional e aplicativos, pode ser usado de maneira *off-line*, ou seja, não conectado a uma rede ou servidor. Em contraste ao *thin client*, ele realiza todo o processamento das aplicações. A figura e a necessidade de um servidor inexistem neste cenário. Os *Thick clients* podem ser chamados de “clientes pesados” já que tudo ocorre localmente.

Dentre as vantagens do *thick client*, pode-se destacar:

- Requisitos mínimos para servidores.
- Desempenho multimídia melhor.
- Maior flexibilidade requer menos largura de banda.
- O cliente tem autonomia, para fazer o processamento e ou verificação dos dados antes de se comunicar com o servidor.

Já dentre as desvantagens dos *thick clients*, pode-se destacar:

- Segurança: servidor deve confiar nos clientes.
- Controle de acesso e autenticação precisa ser gerenciado no servidor.
- Cliente precisa deixar a base de dados do servidor em estado consistente.

### **2.2.3 Principais protocolos utilizados pelas soluções *thin client***

As soluções não-comerciais de *thin clients* utilizam alguns protocolos em comum. Por exemplo, qualquer *thin client diskless* obrigatoriamente necessita de um servidor DHCP para que lhe seja atribuído um IP e, a partir daí, possa acessar a configuração inicial junto ao servidor. Destes protocolos, os principais são o DHCP, O XDMCP, o TFTP, o PXE, o NBD e, para questões voltadas à segurança, o SSH. As seções a seguir apresentam estes protocolos.

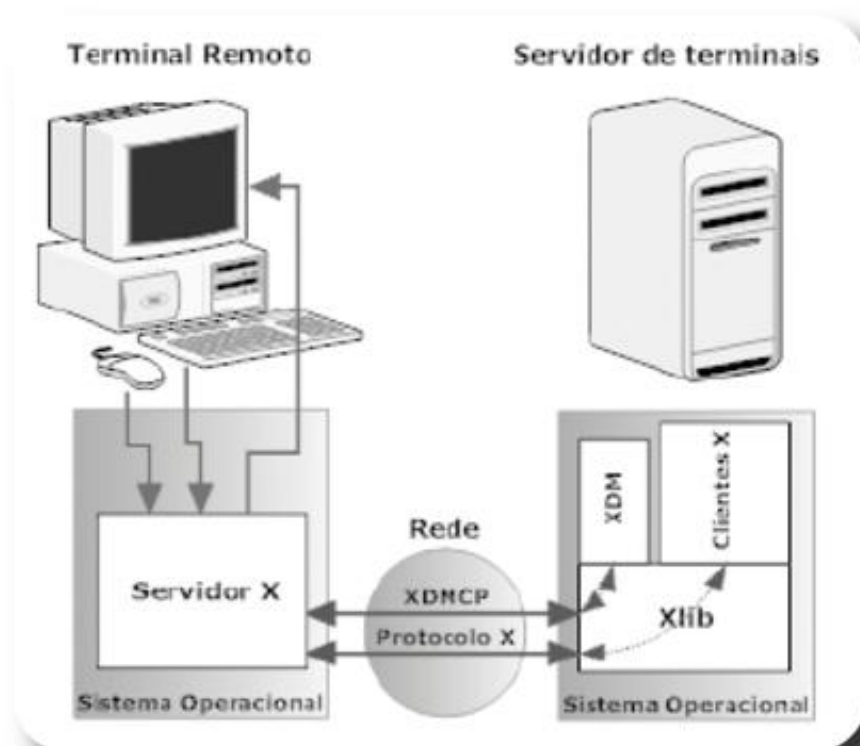
#### **2.2.3.1 XDMCP**

O *X Display Manager Control Protocol* (XDMCP) é um protocolo usado na rede e tem como função fornecer a interface gráfica para os usuários que estão conectados. Ou seja, o usuário mesmo tendo uma máquina mais antiga pode utilizar o ambiente gráfico sem qualquer problema. Por exemplo, um computador compatível com IBM 486 pode utilizar o XDMCP para acessar o ambiente gráfico de um servidor com um processador Intel® Pentium® Dual Core.

Este protocolo utiliza a porta UDP 177 e, quando ativado no servidor; fornece uma saída gráfica para que o usuário que está conectado a rede possa fazer o *login* em sua máquina e assim utilizar todo conteúdo que esta disponível no servidor. Esta é uma maneira de utilizar menos recursos do hardware da máquina do usuário já que todo processamento da interface gráfica é feita pelo servidor.

A Figura 3 demonstra um exemplo do XDMCP em funcionamento.

Figura 3 - Funcionamento do XDMCP



Fonte: Michelin, 2009.

### 2.2.3.2 TFTP

O *Trivial File Transfer Protocol* (TFTP) é um protocolo da camada de aplicação que tem por finalidade a transferência de arquivos, usa a porta 69 do UDP para a transferência de blocos com tamanho fixo de 512 bytes (GAVIDIA, 2012).

Por exemplo, se alguma transferência é iniciada com um pedido de escrita ou leitura de um arquivo ou mesmo para pedir uma conexão, se o servidor reconhece o

pedido a conexão é aberta e o arquivo é enviado ao solicitante num bloco, sendo que se este pacote de dados for menor que 512 *bytes* a transferência é automaticamente cancelada.

Seu acesso não exige autenticação prévia do usuário através do uso de login e senha como o FTP tradicional. Adicionalmente, seu funcionamento não opera com múltiplas conexões, o que o torna menor e mais simples se comparado ao FTP.

Dentro deste processo se um erro acontecer o receptor indicará “*time out*” e fará a retransmissão de seu último pacote, que pode ser de dados ou um reconhecimento. As duas máquinas envolvidas na transferência são consideradas transmissoras e receptoras, já que uma envia dados e recebe reconhecimentos e a outra envia reconhecimentos e recebe dados.

Segundo Gavidia (2012), os erros citados acima podem ser causados por três tipos, apresentados na Tabela 3, conjuntamente com suas possíveis causas.

Tabela 3 - Lista de eventos do TFTP

<b>TFTP – Eventos</b>		
<b>Evento</b>	<b>Descrição</b>	<b>Causa/motivo</b>
<b>1º Evento</b>	Não é possível satisfazer o pedido	<ul style="list-style-type: none"> <li>• Arquivo não encontrado;</li> <li>• Acesso violado;</li> <li>• Não é o usuário.</li> </ul>
<b>2º Evento</b>	Receber um pacote duplicado na rede	<ul style="list-style-type: none"> <li>• Pacote informado incorretamente</li> </ul>
<b>3º Evento</b>	Acesso perdido a um recurso necessário	<ul style="list-style-type: none"> <li>• Disco cheio ou acesso negado durante uma transferência.</li> </ul>

Fonte: Gavidia, 2012.

Os pacotes têm um cabeçalho para permitir transportar-se do meio local de transporte, a ordem do conteúdo de um pacote pode ser:

- Cabeçalho de Meio Local;
- Cabeçalho Internet;
- Cabeçalho Datagrama;
- Cabeçalho TFTP.

A Figura 4 mostra a ordem em que cada cabeçalho aparece.

Figura 4 - Ordem dos cabeçalhos

<b>Meio local</b>	<b>Internet</b>	<b>Datagrama</b>	<b>TFTP</b>
-------------------	-----------------	------------------	-------------

Fonte: Gavidia, 2012

Nesse modelo de cabeçalho, o TFTP acaba não usando o meio local, já que o mesmo é opcional, e também não usa o cabeçalho de *Internet*, utilizando apenas o cabeçalho Datagrama para definir o destino e origem do pacote, bem como o tamanho. Já o cabeçalho TFTP tem como função a identificação do pacote.

O TFTP suporta cinco tipos de pacotes, cujos *Opcodes* (códigos de operação) são:

- 1 – *Read Request* (RRQ): faz a leitura do arquivo de um terminal remoto.
- 2 – *Write Request* (WRQ): envia um arquivo de escrita para um terminal remoto.
- 3 – *Data* (DATA): Faz a confirmação do pacote caso seja de dados.
- 4 – *Acknowledgment* (ACK): faz a confirmação do pacote caso seja de escrita.
- 5 – *Error* (ERROR): informa que algum erro aconteceu na transferência do pacote.

Cada cabeçalho de um pacote TFTP, contém um código de operação (*Opcode*) associado com o pacote, conforme descrito nas figuras que segue.

Nos *Opcodes* 1 e 2 (pacote RRQ e WRQ), o nome do arquivo é uma sequência de bytes terminando num byte zero, conforme Figura 5.

Figura 5 - Pacote RRQ/WRQ

2 bytes	String	1 byte	String	1 byte
<b>Opcode</b>	<b>Filename</b>	<b>0</b>	<b>Modo</b>	<b>0</b>

Fonte: Gavidia, 2012

No *Opcode* 3 (pacote DATA), mostrado na Figura 6, há um número de bloco e um campo de dados.

Figura 6 - Pacote DATA

2 bytes	2 bytes	n bytes
<b>Opcode</b>	<b>Bloco</b>	<b>Data</b>

Fonte: Gavidia, 2012

O *Opcode* 4 (pacote ACK), conforme Figura 7, é reconhecido pelos pacotes DATA e ERROR.

Figura 7 - Pacote ACK



Fonte: Gavidia, 2012

Por fim, o *Opcode* 5 (pacote ERROR), exibido na Figura 8, pode ser reconhecido por algum outro tipo de pacote. O código de erro é um inteiro indicando a natureza do erro.

Figura 8 - Pacote ERROR



Fonte:

No final de uma transferência de arquivos usando o TFTP, ele pode ser designado como:

- **Terminação Normal** - onde ele é marcado por um pacote DATA que contém entre 0 e 511 bytes de dados.
- **Terminação Prematura** - caso o pedido não pode ser reconhecido ou um erro aconteceu durante a transferência é enviado um pacote ERROR.

### 2.2.3.3 PXE

O *Preboot Execution Environment* (PXE) é um ambiente utilizado para realizar a instalação ou a inicialização de Sistemas Operacionais em Computadores, Notebooks através da interface da placa de rede, utilizando os protocolos comuns de rede como IP, UDP, TFTP e DHCP, sem a necessidade dos mesmos possuírem nenhum tipo de armazenamento como HD, Disquete, Cd ou Sistema Operacional já instalado.

Segundo Morimoto (2003):

Este é um padrão de boot remoto desenvolvido pela Intel, que consiste em um pequeno software, gravado na ROM da placa de rede que permite que o PC dê boot através da rede, carregando todo o software necessário a partir de um servidor previamente configurado. (Morimoto, 2003).

Com base nessa afirmação, entende-se que o PXE funciona da seguinte forma: durante o *boot* (inicialização), o computador carrega o PXE, que por sua vez ativa a placa de rede e envia uma solicitação que contém informações de que o cliente possui uma implementação PXE. Essa informação percorre pela rede à procura de um servidor que aceite o tipo de requisição, e enquanto não localizar um servidor disponível, ele percorrerá todos os computadores da rede até localizar um servidor disponível.

Após localizar o servidor, que nesse caso é um servidor DHCP, ele enviará um pacote com todos os dados necessários, como IP, e o caminho necessário do arquivo para que o computador possa começar a carregar o sistema operacional, essas informações ficam armazenadas dentro da *Random Access Memory* (RAM), conhecida como Memória Primária do computador; elas são transportadas utilizando-se o *Trivial File Transfer Protocol* (TFTP). Após o sistema ser carregado, o computador passa a ter o seu funcionamento 100% dependente do servidor e todos os programas e arquivos serão abertos e utilizados diretamente do servidor.

O PXE pode ser usado em ambiente Linux e Windows, dependendo apenas do tipo de ferramenta utilizada. No caso do Linux, pode ser usado o *Linux Terminal Server Project* (LTSP), descrito no decorrer deste Trabalho, e no caso do Windows pode ser usado o *NT Terminal Services*, que possui funcionalidades que permitem o acesso a aplicativos usando um terminal remoto.

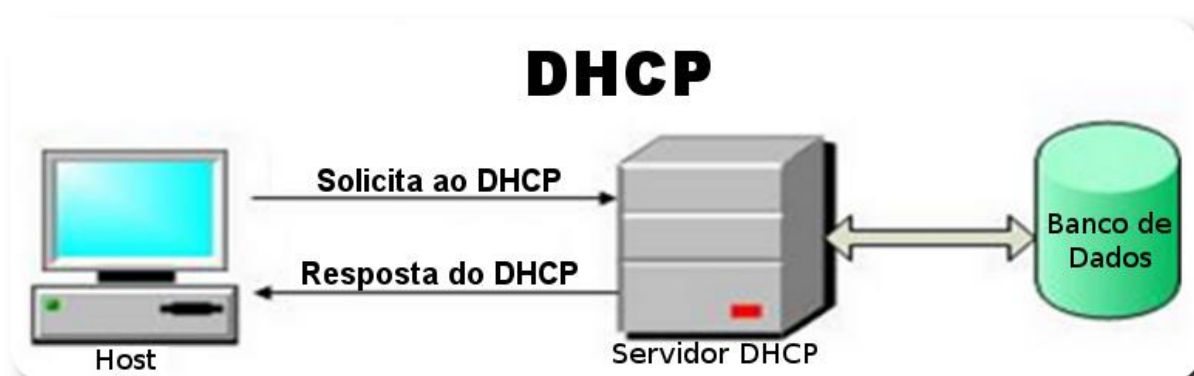
#### **2.2.3.4 DHCP**

O protocolo *Dynamic Host Configuration Protocol* (DHCP) teve seu surgimento no ano de 1983. O mesmo nasceu com o intuito de substituir o *BOOTstrap Protocol* (BOOTP), que tinha por objetivo realizar a função de distribuir configurações automáticas de parâmetro de rede, como o IP e Gateway, para os clientes.

Essa forma de distribuição realizada pelo BOOTP era excelente, porém, ainda havia a necessidade de mudanças, já que o mesmo não alocava os parâmetros dinamicamente. Foi com essa necessidade que nasceu o DHCP.

O DHCP, conhecido também conhecido como Protocolo de Configuração Dinâmica de *host*, tem como objetivo distribuir IP e os demais parâmetros de redes de forma dinâmica para os *hosts*. O seu funcionamento ocorre da forma mostrada na Figura 9.

Figura 9 - Funcionamento do DHCP



Fonte: próprio autor.

Quando um *host* (Computador) solicita permissão para entrar na rede, ele não sabe quem é o servidor DHCP, desta forma ele envia uma solicitação solicitando um IP disponível e os demais parâmetros necessários, do outro lado o servidor verifica se o mesmo se enquadra nos padrões determinados pelo administrador e assim devolver para o *host* os parâmetros necessários, sendo priorizado um IP disponível para que o mesmo possa usufruir da rede dentro das suas limitações, limitações essas que são definidas pelo administrador de rede, quando este Host por algum motivo se desconectar da rede este IP ficará disponível para um novo Host.

Exemplificando de maneira mais simples, pode-se usar como referência o acesso comum à internet, normalmente utilizado em redes residenciais. Nela, é fornecido um IP automaticamente, e toda vez que um usuário acessa a *Internet*, ele estará utilizando um IP diferente, sendo que, uma vez que ele desconecte, estará liberando esse IP, até que outro usuário o utilize.

Nas empresas onde há um grande número de *hosts*, o DHCP facilita muito trabalho dos administradores de rede, onde graças ao DHCP, deixa de ser

necessário que se vá de máquina em máquina realizando todas as configurações necessárias, bastando apenas configurar o servidor DHCP da melhor maneira para que tudo ocorra bem.

Durante a configuração do servidor DHCP, o administrador poderá configurá-lo para que o mesmo possa oferecer três tipos de alocação de endereços IP, sendo elas a manual, automática e dinâmica, conforme segue:

- **Manual:** nessa configuração, associa-se um endereço IP disponível no servidor ao *Media Access Control* (MAC), que corresponde ao endereço da placa de rede de um Host. Assim, o IP disponibilizado para esse MAC não poderá ser utilizado por outro MAC.
- **Automática:** nessa configuração, o endereço IP é associado a um *host* por tempo indeterminado, de forma que quando ele for conectado na rede outra vez, obterá o mesmo IP automaticamente. A diferença entre Automática e Manual é que, na primeira, não é colhido nenhum tipo de informação do *host*.
- **Dinâmica:** nessa configuração o IP é associado a um Host automaticamente por tempo indeterminado e pode ser utilizado por outro Host quando o mesmo estiver disponível. Esta é a configuração que caracteriza o uso do DHCP.

#### 2.2.3.5 NBD

O *Network Block Device* (NBD) é um protocolo baseado na arquitetura cliente/servidor, que emula um dispositivo de bloco, como disco rígido, CD ou disquete, ou seja, qualquer dispositivo que possa transmitir dados em blocos de tamanho fixos, através da rede, passando ao sistema a possibilidade de troca de posições pela rede e até mesmo o espaço em disco já existente.

Em sistemas operacionais como Windows, os dispositivos são denominados como unidades ou drives, seja um disco rígido, CD-ROM e até mesmo *pendrives*. Já no Linux, tudo é tratado como dispositivo. Por esse motivo, no decorrer do texto será mencionado às unidades de discos como dispositivos, por se tratar de um texto voltado para sistemas GNU/Linux.

O NBD oferece para o sistema operacional e seus aplicativos uma ilusão de usar um dispositivo de bloco local, porém, esses dados não estão rodando

localmente e sim no servidor. Isso é possível graças ao controlador de nível de *kernel* chamado de NBD, que se comunica com o servidor NBD via TCP, que por sua vez se encontra configurado para receber essas requisições remotamente.

Assim, sempre que um computador cliente precisar ler algum bloco, ele terá que enviar um pedido ao servidor, que é onde os dados efetivamente estão. Por sua vez, irá responder os dados solicitados, sendo que nesse processo, tudo é tratado por um programa chamado servidor NBD, que tem como propósito apenas atender as solicitações, permitindo o acesso simples ao usuário, pois qualquer navegador de arquivos consegue alcançar o conteúdo do dispositivo. Porém, esse acesso é somente de leitura, os clientes não podem realizar o processo de gravação e as gravações que são feitas pelo cliente em pastas temporárias serão apagadas após o reinício do computador.

Com esse recurso, o NBD favorece o uso de computadores sem discos, ou seja, uma vez que o NBD fornece a ilusão de que os dados estão rodando localmente, os computadores podem utilizar tudo que o servidor oferece, sem a necessidade de possuir um disco rígido, agregando, assim, preços menores e manutenção centralizada dos discos apenas no servidor.

#### **2.2.3.6 SSH**

A sigla SSH vem de Secure Shell, protocolo desenvolvido por Tatu Ylonen, que percebeu que havia a necessidade de uma segurança mais complexa quando teve a sua rede invadida por usar outras ferramentas que possuíam falhas (FERNANDES, 2012).

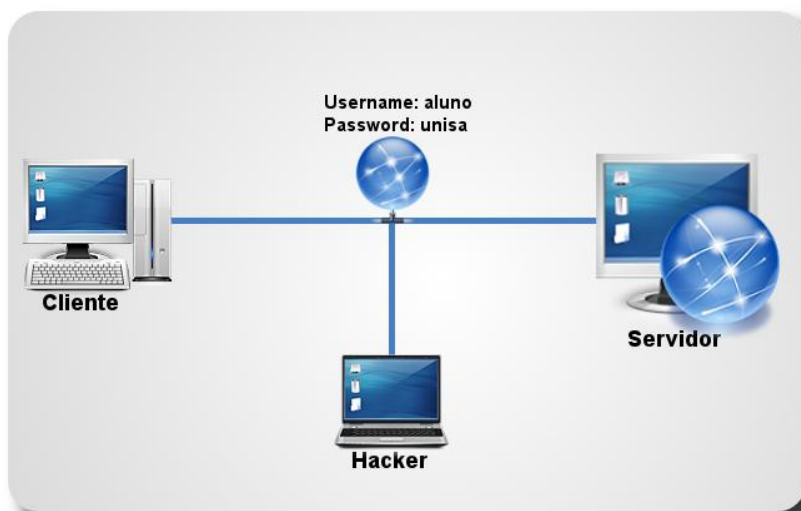
Segundo Morimoto (2003):

*“O SSH é uma espécie de versão evoluída do telnet, que também permite executar arquivos remotamente, mas com várias vantagens”.*

O SSH foi desenvolvido com o intuito de resolver as falhas que as outras ferramentas existentes tinham, visando o acesso remoto mais seguro de um *host*, ferramentas essas como, por exemplo, *telnet* e *rsh*, que já se encontram em

desusos, mas faziam o que prometiam, porém possuíam falhas de segurança, como senhas utilizadas para a conexão, que eram transmitidas em texto puro, sem nenhum tipo de criptografia, facilitando, assim, a sua descoberta por invasores, que apenas “sniffando” a rede no momento em que o administrador iniciava o *login* no servidor, conseguiriam a senha facilmente, conforme mostrado na Figura 10.

Figura 10 - Interceptação de dados em uma rede

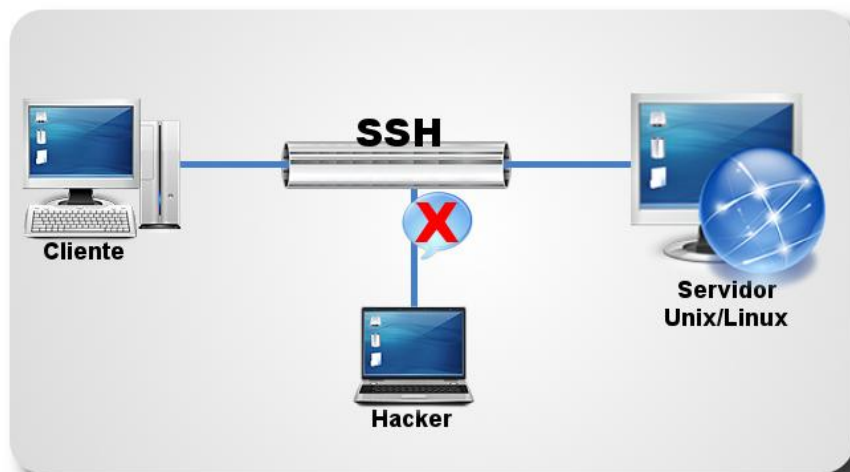


Fonte: Próprio autor

Mas, com o SSH, essa falha foi resolvida da seguinte maneira: todos os dados transferidos na rede são criptografados antes de serem enviados - sejam eles senhas ou arquivos - e ainda dispõem de algumas funcionalidades que diferenciam das outras ferramentas, como por exemplo, proteger os *hosts* de outros tipos de ataques, como *IP spoofing*, *IP sourcerouting* e *DNS spoof*. Dessa forma, os atacantes apenas conseguem parar o serviço, mas não conseguem se tornar proprietário do *host*.

Conforme Pereira (2012), o SSH se encontra rodando na camada de aplicação, e durante o processo de uma conexão remota, são trocadas várias informações em nível de aplicação para prover a encriptação, como a versão do protocolo, os algoritmos e as chaves de codificação. Somente depois dessa troca, todos os dados da camada de aplicação são enviados totalmente encriptados, conforme Figura 11.

Figura 11 - Envio criptografado dos dados usando o ssh



Fonte: próprio autor.

## 2.3 Soluções *thin client*

Existem diversas soluções de *thin clients* disponíveis no mercado, sejam elas open source ou comerciais. Dentre as soluções livres, duas se destacam: o DRBL e o LTSP.

### 2.3.1 LTSP

O LTSP, sigla para Linux Terminal Server Project, é uma solução de *thin client* onde todos os softwares são executados diretamente no servidor, o qual processa as entradas requisitadas pelo cliente e envia, com uso de protocolos específicos, a exibição dos resultados novamente ao cliente, de modo a fornecer um ambiente de trabalho totalmente remoto.

Desenvolvido por Jim McQuillan como uma solução para que um de seus clientes pudesse adicionar trinta e cinco novos usuários à sua rede para uma aplicação de um servidor Unix, o LTSP é distribuído sob a licença GNU GPL, o que garante que a solução seja livre, sendo a mesma distribuída por diversas

distribuições Linux, como RedHat, SuSE e Ubuntu.

Estruturalmente, o LTSP consiste em um conjunto de ferramentas e protocolos com a finalidade de fornecer um ambiente de trabalho remoto para as estações. Essas estações podem ser *thin clients* feitos especificamente para este propósito ou computadores antigos, com baixa capacidade computacional, que podem ser reaproveitados para trabalharem como *thin clients*.

Para seu funcionamento, o LTSP utiliza quatro protocolos distintos, que são:

- 1) **DHCP** – explicado na seção 2.2.3.4, o serviço é necessário para que o *thin client* receba as configurações de rede do servidor;
- 2) **TFTP** – explicado na seção 2.2.3.2, tem a função de transferir o *kernel* a ser utilizado pelos *thin clients*;
- 3) **NBD** – explicado na seção 2.2.3.5, é utilizado para emular o disco rígido a ser utilizado pelos *thin clients*. Em versões anteriores do LTSP, era utilizado o protocolo NFS, nativo dos sistemas baseados em Unix, porém, acabou por ser substituído pelo NBD, que possui um maior controle de permissões.
- 4) **XDMCP** – explicado na seção 2.2.3.1, o XDMCP é o responsável pela exibição da tela de *login* do servidor para acesso remoto do mesmo pelos *thin clients*.

Além desses quatro protocolos, o LTSP utiliza um conjunto de ferramentas próprias, que auxiliam na configuração e no gerenciamento das estações. Essas ferramentas podem ser acessadas através dos seguintes comandos:

- **ltsp-config** – realiza configurações específicas do LTSP em outras ferramentas necessárias para seu funcionamento, como o DHCP. É possível passar ao comando parâmetros de quais arquivos ele irá inserir as configurações necessárias, como por exemplo, os arquivos `lts.conf` ou `isc-dhcp-server`.
- **ltsp-info** – mostra informações sobre o LTSP instalado no sistema, como as versões dos pacotes instalados, versão do sistema operacional ou verificar a existência de arquivos importantes, bem como o local onde se encontram. O resultado do comando `ltsp-info` pode ser visto na Figura 12.

Figura 12 - Resultado do comando ltsp-info

```

root@chiva-hpnetbook: /
root@chiva-hpnetbook:/# ltsp-info
server information:
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 12.04.1 LTS
Release:       12.04
Codename:      precise

server packages:
ii ldm-server 2:2.2.9-1ubuntu0.1
ii ltsp-server 5.3.7-0ubuntu2.2
ii ltsp-server-standalone 5.3.7-0ubuntu2.2
un ltsp-utils <nenhum>
ii ltspfs 1.1-2

packages in chroot: /opt/ltsp/i386
ii ldm 2:2.2.9-1ubuntu0.1
un ldm-themes <nenhum>
ii ldm-ubuntu-theme 2:2.0.47
un ldm-ubuntu-themes <nenhum>
ii ltsp-client 5.3.7-0ubuntu2.2
ii ltsp-client-core 5.3.7-0ubuntu2.2
ii ltspfsd 1.1-2
ii ltspfsd-core 1.1-2

found: /opt/ltsp/i386/etc/lts.conf
found: /var/lib/tftpboot/ltsp/i386/lts.conf
found image: /opt/ltsp/images/i386.img

```

Fonte: próprio autor

- **ltsp-build-client** – comando responsável por gerar uma distribuição de Linux específica para ser usado pelos *thin clients*. É possível especificar parâmetros para definir qual arquitetura a ser utilizada pelos clientes, como por exemplo, i386, definir a pasta base de instalação, dentre outras opções. Ao final, o resultado será, além da pasta contendo os arquivos necessários para iniciar os *thin clients*, um arquivo de imagem a ser utilizado pelo serviço NBD.
- **ltsp-update-image** – atualiza, em caso de alguma alteração, a imagem a ser utilizada pelo NBD.
- **ltsp-localapps** – permite a execução de aplicações de forma a serem processadas pelo próprio *thin client*. O comando é executado no próprio *thin client*, seguido do nome da aplicação a ser executada. A ferramenta é útil para liberar recursos de memória e processamento do servidor, pressupondo que o cliente seja capaz de executá-la. Para isso, faz-se necessário a inclusão do parâmetro LOCAL\_APPS = True na seção default do arquivo lts.conf, além de ser necessário que o comando a ser executado e processado pelo cliente esteja incluso no arquivo de imagem utilizado pelos *thin clients*. Para tanto, utiliza-se o comando **chroot** na pasta gerada pelo

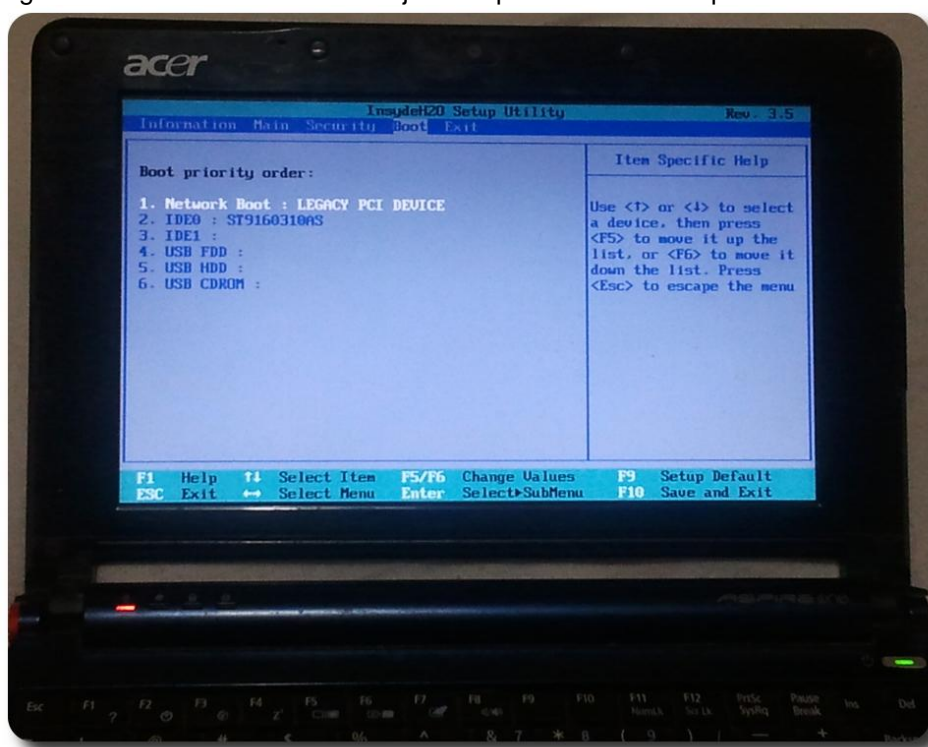
ltsp-build-client, devendo, após a instalação, executar o comando ltsp-update-image para atualizar a imagem a ser utilizada pelo NBD.

Atualmente, o LTSP encontra-se na versão 5.3, embora haja uma grande utilização da versão 4.2 devido aos requerimentos de hardware mais baixos.

### 2.3.1.1 Funcionamento do LTSP

Desde o momento em que um *thin client* é ligado até que se exiba a tela de login, o LTSP executa um conjunto de etapas. A primeira delas utiliza o PXE, responsável pelo boot do sistema via rede. Para tanto, deve-se configurar a BIOS do computador para que o boot seja dado primeiramente a partir da rede, como pode ser visto na Figura 13.

Figura 13 - BIOS de um *netbook* ajustado para realizar boot pela rede

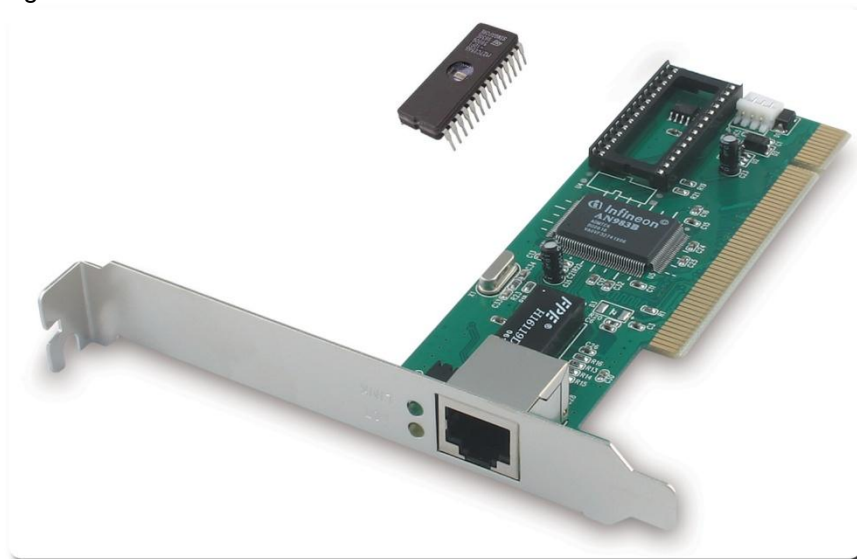


Fonte: próprio autor

O PXE envia um pacote broadcast, aguardando para obter as informações da rede. Caso esta espera seja muito longa, o boot é repassado para o próximo dispositivo definido na BIOS do computador. Existem placas de rede que não

possuem suporte ao PXE, é possível utilizar uma imagem de boot através de um dispositivo removível inicializável, como um CD-ROM, *pendrive* ou disquete, utilizando-se do gPXE, um software para boot em rede *open source*. É possível também utilizar uma EPROM (Figura 14), desde que a placa de rede possua o respectivo encaixe, a qual ficará responsável pelo *boot* pela rede.

Figura 14 - Placa de Rede com *slot* e EPROM



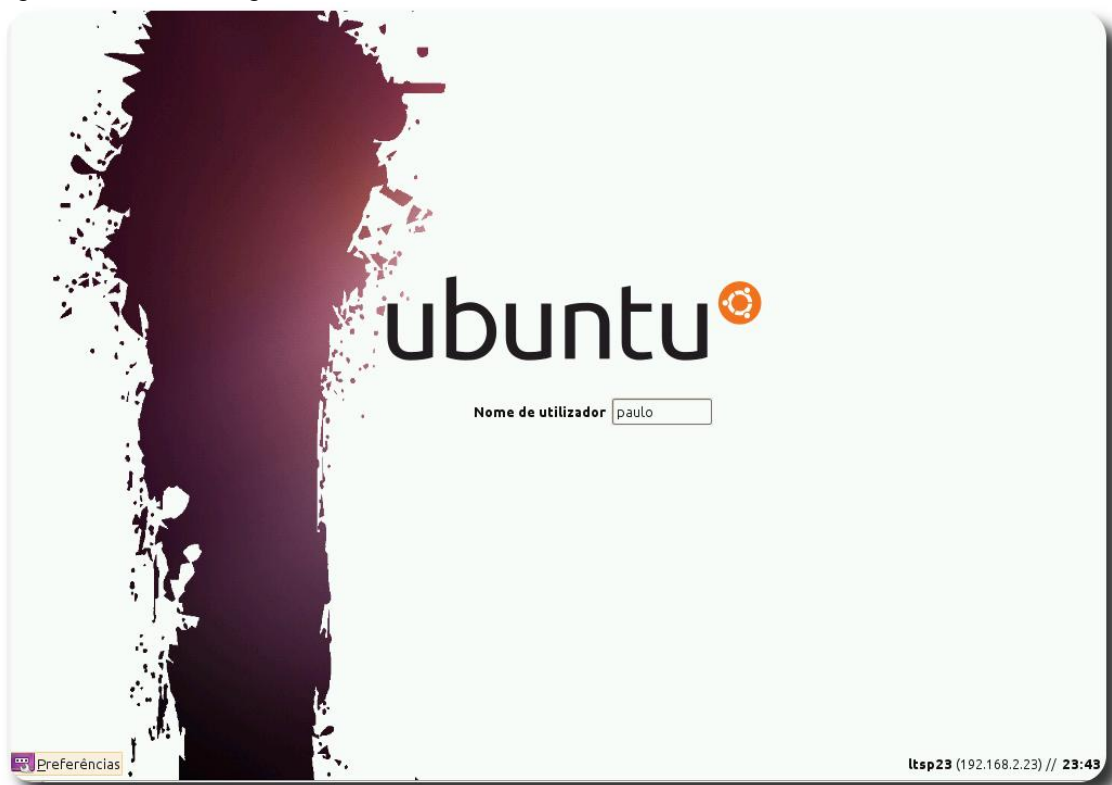
Fonte: ELETRÔNICA, 2012.

O serviço DHCP do servidor, previamente configurado para suportar o PXE, irá fornecer um endereço de rede, bem como a localização da imagem do sistema LTSP, contendo o *kernel* e demais ferramentas necessárias para a realização do boot. Transferência dessa imagem é realizada pelo TFTP, que é a ferramenta adequada para tal função, já que é extremamente simples, não requerendo autenticação e com conjunto de operações reduzido, conforme visto na seção 2.2.3.2.

Uma vez que o *kernel* do sistema é transferido para o cliente, o mesmo passa a ser responsável pelo boot do sistema. Assim, o *kernel* cria um sistema de arquivos virtual, com a utilização do NBD, permitindo a execução de um sistema operacional Linux sem a necessidade de disco rígido ou qualquer outro dispositivo de armazenamento de dados local.

Por fim, é utilizado um tunelamento via SSH para uso seguro do XDMCP, de forma a mostrar a tela de *login* do LTSP, conforme visto na Figura 15.

Figura 15 - Tela de login do LTSP exibida no thin client



Fonte: próprio autor

Como o uso de criptografia acaba por consumir recursos dos *thin clients*, que em geral são equipamentos com baixo poder de processamento, isso pode comprometer o desempenho da estação. Caso a finalidade da rede não necessite de maiores preocupações com segurança, é possível desativar a criptografia, incluindo na seção “default” do arquivo `lts.conf` a linha `LDM_DIRECTX=False`.

Uma vez efetuado o *login*, o usuário terá acesso a todos os aplicativos instalados no servidor, o qual é o responsável pela execução e processamento das informações, exibindo, após, a saída dos resultados na tela do *thin client*.

### 2.3.2 DRBL

O *Diskless Remote Boot in Linux* (DRBL) é uma solução de código aberto utilizada para o gerenciamento e implementação de sistemas operacionais (não precisa necessariamente ser o Linux), em um determinada quantidade de

computadores. A solução, em alguns aspectos, como o método de *boot*, bastante parecida com o LTSP, pois a sua inicialização é feita a partir de um servidor, porém, possui princípios diferentes após o boot (DRBL, 2012).

No caso do DRBL, até mesmo uma máquina normal pode ser utilizada como servidor, já que ele irá apenas fornecer os arquivos solicitados e autenticar os computadores clientes, que por sua vez precisam ser um pouco mais robusto para realizarem as atividades localmente, já que o processamento dos dados não sob responsabilidade do servidor, e sim do próprio cliente.

Essa função até permite a utilização de computadores mais antigos, porém, eles precisam possuir uma configuração compatível com o sistema operacional que a máquina irá utilizar, que não precisa ser necessariamente a mesma do servidor, já que mesmo fica incumbido apenas de fornecer os dados, simulando um dispositivo de blocos, que provido pela rede. Ou seja, todos os processos serão processados localmente no computador após os usuários efetuarem o login, e não no servidor como ocorre no LTSP, e até mesmo HDs locais podem ser utilizados para memória de troca (swap), caso necessário (DRBL, 2012a).

Durante o processo de boot, o computador não precisa possuir o sistema operacional instalado, já que não há necessidade de possuir disco rígido. O processo de boot é feito através do PXE ou *Etherboot*, NFS e NIS, sendo que cada um destes tem função específica e particularidades, mas nada impede que a estação cliente possua um disco rígido com outro sistema operacional, como por exemplo, Windows, para a execução de tarefas que só podem ser realizadas nessa distribuição.

### 3 ESTUDO DE CASO

Este capítulo apresenta o estudo de caso realizado com duas soluções livres para a escolha do software de *thin client* a ser utilizado como solução final na montagem do laboratório apresentado no Capítulo 4. Primeiramente o sistema operacional utilizado é apresentado (Seção 3.1). A seguir, detalha-se os equipamentos utilizados neste estudo de caso (Seção 3.2). Na sequência, demonstra-se o processo de instalação e configuração das soluções LTSP (Seção 3.3) e DRBL (Seção 3.4). A Seção 3.5 mostra a análise de desempenho realizada nas duas soluções, a Seção 3.6 mostra a análise dos resultados obtidos e este capítulo termina com a Seção 3.7 definindo a solução escolhida para implementação.

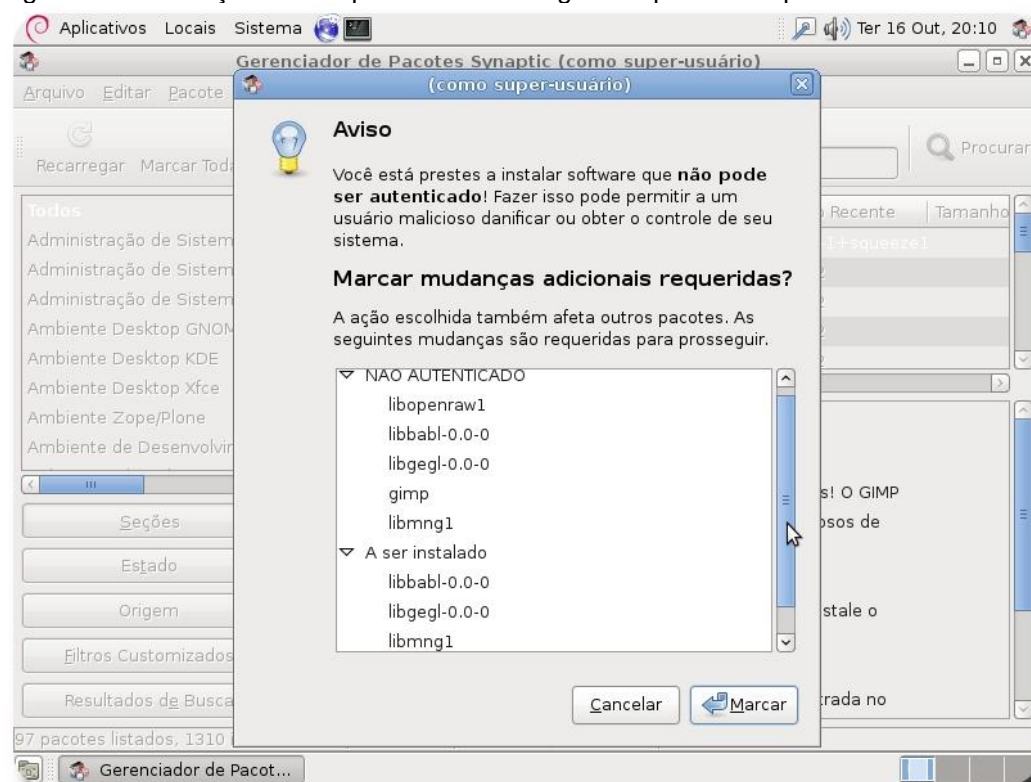
#### 3.1 Sistema operacional utilizado

As duas soluções foram aplicadas em um ambiente de testes utilizando-se como sistema operacional o Ubuntu 12.04. O Ubuntu, dentre diversas distribuições existentes no mundo GNU/Linux, mostrou-se a opção mais adequada devido a dois fatores. Primeiro porque, dentre todas as grandes distribuições, é a que foca seu desenvolvimento na interação entre o usuário final, aquele sem conhecimentos mais aprofundados de informática, ou seja, a distribuição busca facilitar ao máximo diversas tarefas comuns, que vão desde a instalação de impressoras até a instalação facilitada de aplicativos.

Apenas a título de comparação, para instalar o aplicativo de imagens Gimp na distribuição GNU/Linux Debian, utilizando-se a interface gráfica padrão, é necessário clicar-se no menu “Sistemas”, “Administração” e Gerenciador de Pacotes Synaptics. Uma vez aberto o aplicativo, o usuário deve pesquisar pelo aplicativo Gimp. Na lista de resultados, deverá clicar com o botão direito do mouse no aplicativo Gimp listado, e clicar em “marcar para instalação”. Após, aparecerá uma mensagem listando todas as dependências necessárias para a instalação do aplicativo (Figura 16), o que o usuário deverá confirmar, mesmo que não saiba exatamente para que servem estes

pacotes listados. Uma vez confirmada a instalação das dependências, o usuário deverá clicar no botão “aplicar”, para finalmente ter a aplicação instalada, que aparecerá no menu de aplicações. Já utilizando o Ubuntu, o usuário deverá clicar no botão do aplicativo “Central de Programas” que já se encontra no *dash*, a barra lateral de aplicações da interface gráfica. Após, deverá pesquisar pelo Gimp na barra de busca, que retornará a lista de resultados. Na lista, basta clicar no botão “Instalar” que aparecerá ao lado do resultado. O aplicativo será instalado, sem qualquer questionamento, bem como irá adicionar um atalho no *dash*, facilmente identificado com o ícone do aplicativo.

Figura 16 - Instalação do Gimp no Debian: listagem de pacotes dependentes

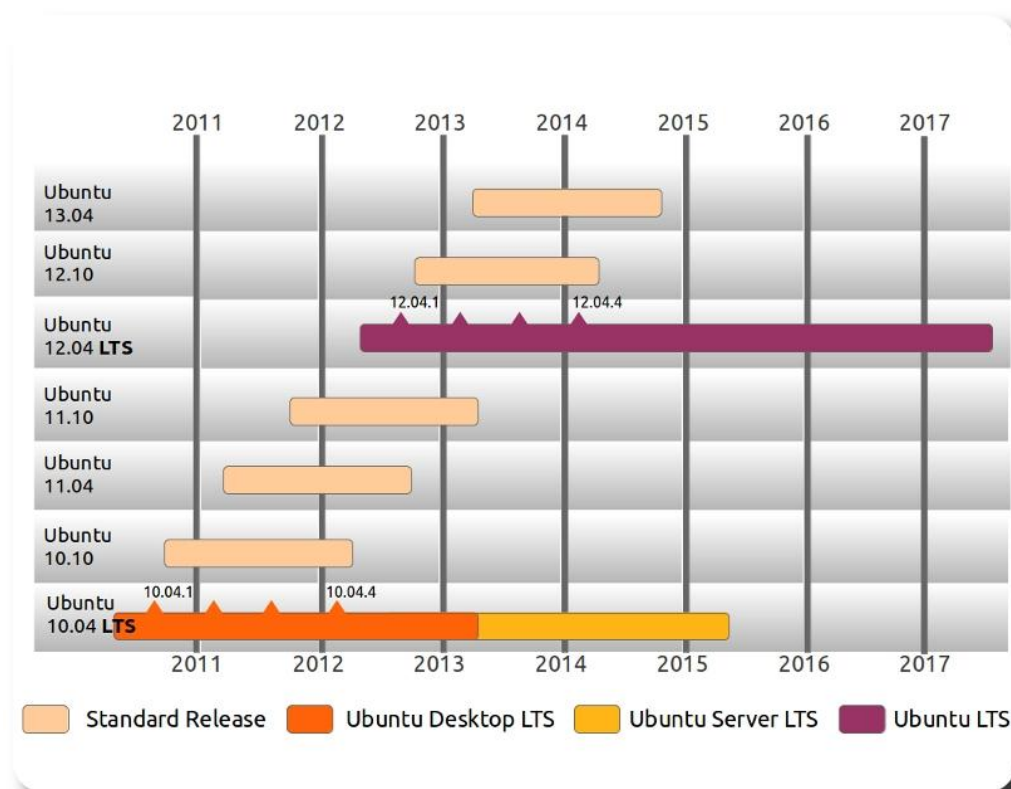


Fonte: próprio autor.

O segundo fator que torna o Ubuntu a distribuição adequada para o uso em redes *thin clients* é o fato de ter um ciclo bem definido de suporte aos aplicativos e desenvolvimento de novas versões. Uma nova versão do Ubuntu é lançada em ciclo de seis meses; o próprio controle de versões é baseado neste ciclo. Por exemplo, a versão 12.04 nada mais é do que a versão lançada no ano 12, mês 04, ou seja, em abril de 2012. A versão mais recente foi lançada em 18 de outubro de 2012. Logo,

de acordo com o controle de versão adotado, ela é chamada de Ubuntu 12.10. Quanto ao suporte, a Canonical, responsável pela distribuição, trabalha em duas frentes, conforme pode ser observado na Figura 17, onde é lançada uma versão com o que há de mais recente, porém, com período de suporte mais curto, de um ano e seis meses; e outra versão, denominada pela Canonical como LTS, sigla para *Long Term Support*, onde a empresa busca lançar uma versão com maior preocupação contra falhas de segurança e *bugs* em geral, deixando de incluir versões recentes de aplicativos caso estes não sejam considerados estáveis, porém, o suporte com atualizações de segurança. Na versão LTS o suporte é de cinco anos, consideravelmente maior que as demais versões. Assim, o usuário da distribuição Ubuntu, versão 12.04 LTS, terá suporte a atualizações de segurança até abril de 2017.

Figura 17 - Ciclo de lançamento e suporte do Ubuntu



Fonte: Stewart, 2012.

O Ubuntu é distribuído com um conjunto de aplicativos voltados às tarefas mais comuns realizadas pelos usuários, como gerenciamento de arquivos, navegação de Internet, envio e recebimento de e-mails, gerenciamento de fotos,

imagens áudio e vídeo. Logo, após a instalação do sistema, não há a necessidade de instalar qualquer outro aplicativo, cabendo ao administrador da rede instalar algum de uso específico.

A próxima seção apresenta o hardware utilizado nos testes para escolha da solução LTSP a ser utilizada.

### 3.2 Equipamentos utilizados para o estudo de caso

Para que fosse feita a análise de desempenho das soluções LTSP e DRBL, foram utilizados os seguintes equipamentos:

- Um notebook, que trabalhou como servidor, possuindo as configurações descritas na Tabela 4;
- Dois notebooks que trabalharam como estações *thin client*, possuindo as configurações mostradas na Tabela 4. Na realização dos testes, as estações serão chamadas de TC01 e TC02;
- Um switch de oito portas, da Marca Tenda, modelo S8, conforme visto na Figura 18;
- Três cabos de rede UTP categoria 5, para conexão dos computadores ao switch.

Figura 18 - *Switch* de oito portas utilizado nos testes



Fonte: TENDA, 2012.

No servidor, o disco rígido foi fracionado em partições distintas: uma contendo o sistema operacional configurado para uso do LTSP, conforme seção 3.3 deste

capítulo, e outra partição contendo o sistema operacional configurado para uso do DRBL, conforme descrito na seção 3.4. Nos *thin clients*, os discos rígidos foram desabilitados, já que não será feito uso dos mesmos.

Tabela 4 - Especificações do Servidor e *thin clients*

ESPECIFICAÇÕES			
Configuração	Servidor	TC01	TC02
Processador	AMD E350 1.6Ghz	Atom N280 1.66Ghz	i3 350M 2,26Ghz
Memória	2GB DDR3	2GB DDR2	2GB DDR3
Vídeo	Radeon HD 6310	Intel 945	Intel HD Ironlake M
Rede	Realtek RTL8111	RealtekRTL8102e	Realtekpci-e FE

Fonte: próprio autor

### 3.3 Instalando e Configurando o LTSP

Depois de instalado o Ubuntu 12.04, a instalação e configuração do LTSP são realizadas em doze passos distintos. Para a instalação dos pacotes e a subsequente configuração, é necessário privilégios de administrador, para o qual utiliza-se o comando **sudo**, que nada mais é do que um comando que permite usuários comuns obtenham privilégios do usuário *root*. Os passos necessários são:

- 1) Na primeira etapa, é feita a atualização da lista de pacotes do Ubuntu, realizada através do comando **sudo apt-get update**.
- 2) Uma vez que a lista é atualizada, é feita a atualização dos pacotes propriamente dita, através do comando **sudo apt-get upgrade**.
- 3) Com os pacotes devidamente atualizados, o próximo passo é a instalação dos pacotes *ltsp-server-standalone* e o pacote *openssh-server*, este último necessário para que os clientes realizem o tunelamento via *ssh* ao para obtenção do ambiente gráfico (X Server) junto ao servidor LTSP. Para o correto funcionamento destes dois pacotes, *apt* instala também pacotes adicionais para o funcionamento do serviço, que são:
  - **debconf-utils** – pacote que contém pequenos utilitários para desenvolvedores do *debconf*, que é o sistema de gerenciamento de configuração do Debian, distribuição de Linux no qual o Ubuntu é

baseado;

- **debootstrap** – É utilizado para criar um sistema básico do zero, baixando os arquivos necessários do site;
- **fuse-utils** – É utilizado para exportar um sistema de arquivos virtual para o *kernel*;
- **isc-dhcp-server** – É um pacote utilizado para atribuir endereços IP aos hosts conectados na rede e todas as configurações de rede necessários para o host cliente;
- **Idm-server** – É um gestor de tela do X11 que utiliza o túnel SSH para disponibilizar um mecanismo de *login* para as sessões remotas;
- **liblz2-2** – É uma biblioteca que realiza compressão e descompressão de dados de forma muito rápida, sendo que a descompressão não requer memória.
- **tsp-server** – É um protocolo responsável por negociar túnel IP entre um cliente e o servidor TSP;
- **Itspfs** – É um sistema de arquivos remoto que utiliza um servidor *daemon* para conexões entre o cliente e o servidor ;
- **nbd-server** – É um protocolo cliente servidor que emula um dispositivo de bloco como um HD, disquete através da rede;
- **openbsd-inetd** – OpenBSD *Internet Superserver*, programa especializado em gerenciar conexões recebidas pela rede;
- **pulseaudio-esound-compat** – É um servidor de som para sistemas win32, e contém os componentes de compatibilidade para versões mais antigas de desktops;
- **squashfs-tools** – É um sistema de ficheiro para Linux altamente comprimido sendo somente de leitura;
- **ssh-import-id**- Recupera uma ou mais chaves públicas de um servidor público e anexa ao processo do usuário atual;
- **tftpd-hpa** – É utilizado para transferência de arquivos através da rede para outros clientes de uma rede no caso do PXE;
- **xbase-clients** – É programa responsável por gerar a interface com um servidor X utilizando as bibliotecas do X utilizando os dispositivos de entrada e saída como monitores, placa de vídeo e teclado.

Para a instalação dos pacotes e todas as suas dependências, utiliza-se o comando **sudo apt-get install ltsp-server-standalone openssh-server**.

- 4) O próximo passo é informar ao DHCP qual a interface de rede que receberá as requisições dos clientes, neste caso, a interface eth0. Para isso, deve-se editar o arquivo /etc/default/isc-dhcp-server e incluir uma linha com o texto:

```
INTERFACES="eth0"
```

Para edição do arquivo em questão utilizou-se o editor vi, através do comando **sudo vi /etc/default/isc-dhcp-server**, acrescentando-se a já mencionada linha.

- 5) A seguir, deve-se atribuir o IP a ser utilizado na interface de rede do servidor. Para isso, deve ser editado o arquivo /etc/network/interfaces, que pode ser editado pelo vi através do comando **sudo vi /etc/network/interfaces**. Neste estudo, utilizou-se o IP 192.168.2.1 com máscara de sub-rede 255.255.255.0 para o servidor, que, no arquivo em questão, ficou como mostrado abaixo:

```
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet static
address 192.168.2.1
netmask 255.255.255.0
```

- 6) Uma vez configurada a interface de rede, a mesma deve ser iniciada através do comando **sudo ifup eth0**.
- 7) O passo seguinte é ajustar o arquivo de configuração do dhcp utilizado pelo ltsp. Para editá-lo, pode ser utilizado o seguinte comando: **sudo vi /etc/ltsp/dhcpd.conf**. Neste arquivo todas as faixas de IP (no LTSP, o padrão é 192.168.0.0/24) devem ser substituídas pela mesma utilizada pela interface de rede, ou seja, 192.168.2.0/24. Ao final, o arquivo ficará como abaixo:

```
authoritative;
subnet 192.168.2.0 netmask 255.255.255.0 {
    range 192.168.2.20 192.168.2.250;
    option domain-name "example.com";
    option domain-name-servers 192.168.2.1;
    option broadcast-address 192.168.2.255;
    option routers 192.168.2.1;
```

```

option subnet-mask 255.255.255.0;
option root-path "/opt/ltsp/i386";
if substring( option vendor-class-identifier, 0,
9 ) = "PXEClient" {
    filename "/ltsp/i386/pxelinux.0";
} else {
    filename "/ltsp/i386/nbi.img";
}
}

```

A configuração utilizada neste trabalho atribuiu os IPs dinamicamente, não sendo necessário acrescentar outras linhas no arquivo de configuração.

- 8) Após configurado o dhcp utilizado pelo LTSP, deve-se reiniciar o serviço `isc-dhcp-server`, através do comando **`sudo service isc-dhcp-server restart`**.
- 9) Após iniciado o servidor dhcp, as chaves ssh utilizadas para acesso dos clientes ao servidor devem ser atualizada com o comando **`sudo ltsp-update-ssh keys`**.
- 10) Uma vez configurados os serviços necessários para funcionamento do LTSP, bem como as interfaces de rede, o próximo passo foi criar a imagem que será carregada pelo cliente, através do comando **`sudo ltsp-build-client --arch i386`**. O comando irá baixar os pacotes necessários para o carregamento *thin client*, como *kernel*, servidor X, entre outros e, ao final, utilizará o `squashfs-tools` (instalado como pacote adicional durante a instalação do pacote `ltsp-server-standalone`) para criação do arquivo de imagem. Ao final do processo será exibida a mensagem "info: instalação do cliente LTSP finalizada com sucesso".
- 11) Com a imagem a ser utilizada pelos *thin clients* criada, deve-se reiniciar o serviço `nbd-server`, responsável pela emulação de dispositivos de blocos, neste caso, um disco rígido. Para reiniciar o serviço, utiliza-se o comando **`sudo service nbd-server restart`**.
- 12) Por fim, deve-se editar o arquivo `var/lib/tftpboot/ltsp/i386/ltsp.conf` para configurar o comportamento dos *thin clients*. Na configuração proposta, o único ajuste é relacionado ao gerenciador de janelas a ser utilizado nas estações. Por padrão, o Ubuntu utiliza o Unity, um *shell* para o ambiente

Gnome. Porém, como a configuração original utiliza um ambiente com aceleração gráfica, isso inviabiliza o carregamento de forma correta nas estações, que, por geralmente tratarem-se de equipamentos obsoletos, não possuem placas de vídeo com aceleração em 3D. Para resolver este problema, ao invés de utilizar o Unity padrão, deverá ser utilizado Unity2D, que já vem previamente instalado na versão 12.04 do Ubuntu, objeto deste trabalho. Para isso, primeiro deve ser digitado o comando **sudo vi /var/lib/tftpboot/lts/i386/lts.conf** para edição do arquivo de configuração, que estará em branco, e adicionar as seguintes linhas:

```
[default]
LDM_DIRECTX=True
LDM_SESSION="gnome-session --session=ubuntu-2d"
```

Depois de seguidos os doze passos descritos, o servidor estará pronto para receber os *thin clients*.

### 3.4 Instalando o DRBL

Antes da instalação e configuração do DRBL, é necessário realizar alguns ajustes no servidor. Primeiramente, o script de instalação, em determinado ponto, exige que se tenham duas placas de rede distintas: uma para uso pelo DRBL, e a outra para os demais, como por exemplo, Internet, e não é possível que a placa seja wireless. Caso o servidor possua apenas uma placa de rede com fio, é possível criar uma “placa virtual” simulando duas placas distintas com o uso de apenas uma. Para tanto, deve-se utilizar o comando **sudo ifconfig eth0:1 192.168.1.1/24 up**. Com isso, o instalador do DRBL interpretará como se houvesse duas placas de rede distintas: a eth0, a ser utilizada pelo DRBL, e a eth0:1, para uso, neste caso, do compartilhamento da Internet.

No manual de instalação do DRBL também há a recomendação para que seja desinstalado o serviço Network Manager, através do comando **sudo apt-get remove network-manager**. Porém, em todos os testes realizados neste trabalho, o serviço foi mantido sempre ativo, não causando qualquer problema ao funcionamento do DRBL.

Diferentemente do LTSP, o DRBL não possui suporte padrão pelo Ubuntu, não havendo pacotes de instalação nos repositórios. Devido a isso, inicialmente, deve-se adicionar os repositórios oficiais da solução à lista de repositórios do apt. Para isso, é necessário editar o arquivo de lista de fontes do apt, qual seja, `/etc/apt/sources.list`, através do comando **sudo vi /etc/apt/sources.list** e adicionar as duas linhas que segue:

```
deb http://archive.ubuntu.com/ubuntu precise main
restricted universe multiverse
deb http://drbl.sourceforge.net/drbl-core drbl stable
```

Após adicionado o repositório na lista de repositórios do apt, deve-se instalar as chaves de segurança do repositório, através do comando **wget -q http://drbl.org/GPG-KEY-DRBL -O- | sudo apt-key add -**. Caso a chave não seja instalada, quando da instalação do DRBL, será exibido alerta de instalação sem verificação, conforme Figura 19, exigindo que o usuário confirme a instalação dos pacotes.

Figura 19 - Instalação do DRBL sem a chave de assinatura

A terminal window titled 'paulo@chiva-hp-netbook: ~' showing the command 'sudo apt-get install drbl'. The output indicates that the package 'drbl' is being installed from a source that cannot be authenticated. The terminal text is as follows:

```
paulo@chiva-hp-netbook:~$ sudo apt-get install drbl
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
Os NOVOS pacotes a seguir serão instalados:
 drbl
0 pacotes atualizados, 1 pacotes novos instalados, 0 a serem removidos e 0 não a
tualizados.
É preciso baixar 2.397 kB de arquivos.
Depois desta operação, 6.943 kB adicionais de espaço em disco serão usados.
AVISO: Os pacotes a seguir não podem ser autenticados!
 drbl
Instalar estes pacotes sem verificação [s/N]?
```

Fonte: próprio autor.

Com a chave de assinatura devidamente configurada, o próximo passo é atualizar a atualizar a lista de pacotes, o que deve ser feito através do comando **sudo apt-get update**.

Uma vez atualizada a lista de pacotes, deve-se instalar o DRBL propriamente dito, através do comando **sudo apt-get install drbl**.

Uma vez instalados os pacotes, o passo seguinte é a execução do script de configuração do servidor DRBL. Para tanto, deve-se digitar o comando **sudo /opt/drbl/sbin/drblsrv -i**. A partir da execução do comando, a configuração se dá em quatro passos distintos.

- 1) No primeiro passo é dada a opção de instalar algumas imagens de boot de distribuições Linux populares, sendo exibido o texto que segue:

```
*****.
Instalando DRBL para Debian Linux...
*****.
Deseja instalar as imagens de inicialização (boot) de rede que
permitem que o cliente instale alguma distribuição GNU/Linux
(Debian, Ubuntu, RedHat Linux, Fedora Core, Mandriva, CentOS e
OpenSuSE...) através da rede? ///NOTA/// Esta escolha abaixará
muitos arquivos (> 100 MB no total) da Internet, portanto
levará alguns minutos. Se seu cliente possui disco rígido e é
possível instalar o GNU/Linux nele, digite Y aqui. Se você
colocar "no" aqui, pode executar drbl-netinstall para instalá-
los depois.
[y/N]
```

Como as *estações thin clients* utilizarão a mesma distribuição de Linux usada no servidor, não há a necessidade de imagens de instalação, devendo simplesmente ser pressionada a tecla Enter para escolher a opção padrão "N".

- 2) No passo seguinte, o usuário é questionado sobre o uso de console serial:

```
*****.
This GNU/Linux distribution uses one kernel to support SMP and
non-SMP arch.
*****.
Deseja usar a saída de console serial para os clientes?
Se você NÃO sabe do que se trata, digite "N" aqui, senão os
clientes mostrarão NOTHING (NADA) na tela !
[y/N]
```

Como o objetivo da rede *thin client*, no caso deste trabalho, é oferecer um ambiente gráfico para acesso a aplicativos de uso comum, não há necessidade de utilização de consoles seriais, devendo ser pressionada a tecla Enter para novamente escolher a opção padrão "N".

- 3) O próximo passo não exige interação do usuário, dando apenas o aviso relacionado à disponibilização do *kernel* utilizado pelo servidor, conforme segue:

```
*****.
This server is running Ubuntu Linux, so generic kernel is
available.
*****.
Está disponível apenas o mesmo tipo de kernel que o servidor
para os clientes no repositório.
Pressione "Enter" para continuar...
```

Pressionando a tecla Enter, é dado a mensagem informando que a otimização do sistema será configurada para a mesma que a do servidor:

```
The CPU arch option for your clients: 2
A otimização de seu sistema está configurada para ser a mesma
deste servidor.
*****
```

A partir de então, é realizada a atualização da lista de repositórios do servidor, perguntando ao final se o usuário deseja atualizar o sistema operacional:

```
*****.
Deseja atualizar seu sistema operacional?
[y/N]
```

A atualização refere-se ao sistema operacional utilizado no servidor. É recomendado que a atualização seja feita neste momento, de forma a evitar que uma atualização imediatamente após a configuração do DRBL resulte em arquivos de configuração e scripts sobrescritos pela instalação dos novos pacotes, resultando em falhas no funcionamento. Este é um cuidado que deva ser tomado sempre que grandes atualizações do sistema sejam realizadas. Logo, deve-se pressionar a tecla “y”, seguida da tecla Enter. Após a atualização do sistema, o script passa a instalar os pacotes e ferramentas (ou atualizá-los, caso já se encontrem instalados no servidor) necessários para funcionamento do DRBL.

- 4) A seguir, é questionado sobre qual *kernel* será utilizado pelos *thin clients*, sendo dadas duas opções, conforme segue:

```
*****.
Tentando atualizar alguns pacotes se for necessário...
*****.
No repositório ayo, procurando o mais recente kernel ...
The kernel image in Ubuntu 12.04 "uses generic" for i686/amd64
CPU.
O kernel mais recente do repositório ayo é linux-image-3.2.0-
32-generic
Há 2 kernels disponíveis para os clientes, qual você prefere?
[1]: kernel 3.2.0-32-generic-pae (a partir deste servidor DRBL)
```

```
[2]: linux-image-3.2.0-32-generic (a partir do repositório apt)
[1]
```

A primeira opção oferece o *kernel* encontrado no servidor de pacotes do DRBL e a segunda, o encontrado no próprio repositório do Ubuntu 12.04. Ambas as opções trazem o mesmo resultado final, já que se trata da mesma versão do *kernel*, podendo o usuário escolher qualquer das opções. Feita a escolha, é feito o download do *kernel*, gerado os arquivos de configuração do PXE e os arquivos de imagem para os clientes PXE e Etherboot.

Concluída a instalação do DRBL, a próxima etapa é configurar os *thin clients* junto ao servidor. Para isso, deve-se executar o comando **sudo /opt/sbin/drblpush -i**. O comando nada mais é que a execução de um script para configuração, conforme passos a seguir:

- 1) A primeira etapa questiona qual o domínio DNS e o nome NIS/YP a ser utilizado, conforme segue:

```
*****
Procurando os pacotes instalados para o servidor DRBL... Isto
pode demorar vários minutos...
Concluída a pesquisa dos pacotes instalados para o servidor
DRBL.
*****
-----
O modo interactivo permite a você fornecer a informação de seu
ambiente DRBL.
-----
-----
Por favor introduza domínio DNS (como drbl.sf.net):
[drbl.name]
Por favor introduza o nome NIS/YP do domínio:
[penguinzilla]
```

Como não há o que ser configurado, basta apertar a tecla Enter em cada uma das perguntas. Logo após, é solicitado o *hostname* a ser dado aos clientes:

```
-----
Por favor introduza o prefixo do hostname cliente:
Este prefixo será usado automaticamente para criar hostnames
para os clients. Se deseja sobrescrever algum ou todos os
hostnames criados automaticamente, pressione Ctrl-C para sair
deste programa agora, edite /opt/drbl/conf/client-ip-hostname,
e execute este programa outra vez.
[drbl]
```

Após preenchido o hostname, basta apertar a tecla Enter para ir ao próximo passo.

- 2) A seguir, é perguntado qual interface de rede será utilizada para acesso público à Internet:

```

-----
eth0: IP address 192.168.2.1, netmask 255.255.255.0
eth1: IP address 192.168.43.230, netmask 255.255.255.0
Placa(s) de rede configurada(s) encontradas em seu sistema:
eth0 eth1
-----

```

```

Não foi encontrado o endereço IP público deste servidor.
Qual porta ethernet deste servidor é para acesso público à
Internet, não para conexão DRBL ?
Portas ethernet disponíveis neste servidor:
eth0 (192.168.2.1), eth1 (192.168.43.230),
[eth1]

```

Como, assim como no LTSP, será utilizada a interface eth0 para o tráfego de dados dos *thin clients*, a interface disponível para acesso à Internet será a eth1.

- 3) A seguir, é feita a configuração dos IPs utilizados pelos *thin clients*. Inicialmente, o DRBL pergunta se o usuário deseja que sejam coletados os endereços MAC dos clientes:

```

*****
*****
Agora podemos coletar os endereços MAC dos clientes!
Se deseja que o serviço DHCP no servidor DRBL ofereça o mesmo
endereço IP ao cliente toda vez que o cliente inicie, e nunca
fez este procedimento, faça-o agora!
Se já tem estes endereços MAC dos clientes, pode pô-los em
diferentes arquivos de grupo (O número de arquivos é o mesmo
número de placas de rede do serviço DRBL). Neste caso, pode
pular este passo.
Este passo o ajuda a salvar os endereços MAC dos clientes, e os
divide em diferentes grupos. Economiza tempo e reduz erros.
Os endereços MAC serão salvo um por um segundo a inicialização
dos clientes,
e serão postos em diferentes arquivos segundo a placa de rede
do servidor, o nome do arquivo será como macadr-eth1.txt,
macadr-eth2.txt... Pode encontrá-los no diretório /etc/drbl.
Por favor inicie os clientes por ordem, assegure-se de que
iniciem a partir do etherboot ou PXE!
deseja coletá-los ?
[y/N]

```

Caso o usuário confirme, será solicitado que os *thin clients* sejam iniciados, um a um, para que sejam capturados os endereços MAC de suas interfaces de rede, realizando ajustes no DHCP para que lhes sejam atribuídos endereços de IP fixo. Como se pretende que a configuração dos equipamentos seja a mais simples possível, deve-se pressionar a tecla N, seguido de Enter para partir para o próximo passo, fazendo com que o servidor DHCP do DRBL atribua os IPs dinamicamente. Para isso, após confirmada a opção anterior, aparecerá a seguinte opção:

```

*****

```

Deseja que o serviço DHCP no servidor DRBL ofereça o mesmo endereço IP ao cliente cada vez que o cliente inicie (Se deseja esta função, deve coletar os endereços MAC dos clientes, e salvá-los em arquivo(s) (como no procedimento anterior)). Isto é para os clientes conectados à interface de rede ethernet do servidor DRBL eth0 ?

[y/N]

Assim, deve-se digitar N e pressionar Enter para, finalmente, escolher a faixa de IP a ser atribuída aos *thin clients*, conforme as duas opções a seguir:

```
*****
OK! Continuemos, configurando o endereço IP dos clientes por
"primeiro a iniciar recebe o IP primeiro" em vez de fixá-los!
Host min: 192.168.2.1
*****
Qual é o número inicial que deseja usar no último grupo de
dígitos do endereço IP (por ex. o valor inicial de d no
endereço IP a.b.c.d) dos clientes DRBL conectados a esta porta
ethernet eth0.
[1]

*****
Quantos clientes DRBL (PC para estudantes) estão conectados à
interface de rede ethernet do servidor DRBL eth0 ?
Por favor introduza o número:
[12]
```

Será definido um padrão máximo de vinte *thin clients*. Logo, na primeira opção, deve-se digitar 1 e, na segunda 20. Feito isto, será solicitada a confirmação da faixa de IP e, logo após, apresentado um *layout* de como será a rede:

```
*****
OK! Continuemos...
*****
O Layout de seu ambiente DRBL:
*****
          NIC      NIC IP                      Clients
+-----+-----+-----+
|          DRBL SERVER          |
|          |          |          |
|    +-- [eth1] 192.168.43.230 +- to WAN |
|          |          |          |
|    +-- [eth0] 192.168.2.1 +- to clients group 0 [ 20 clients,
their IP
|                                     |from 192.168.2.1 - 192.168.2.20]
+-----+-----+-----+
*****
Total clients: 20
*****
Pressione Enter para continuar...
```

- 4) Feita a configuração dos IPs dos *thin clients*, o DRBL configurará o modo como o serviço irá se comportar. Os modos disponíveis, devidamente explicados, são exibidos para seleção pelo usuário:

```

-----
No sistema, há 3 modos para os serviços Linux diskless:
[0] Modo DRBL completo, cada cliente tem seu próprio NFS
baseado em /etc e /var.
[1] Modo DRBL SSI (Single system image), cada cliente usa tmpfs
baseado em /etc e /var. Neste modo, a carga e o espaço de disco
necessário do servidor será mais leve. NOTA! (a) Se recomenda
pelo menos 256 MB de memória na máquina cliente. (b) A
configuração e os arquivos de configuração do cliente não serão
salvos no servidor DRBL! Serão usados apenas uma vez e
descartados quando a máquina for desligada! Além disso, se
modificar algum arquivo do cliente modelo (localizado em
/tftpboot/nodes), deve executar /opt/drbl/sbin/drbl-gen-ssi-
files para criar a tarball modelo em
/tftpboot/node_root/drbl_ssi/. (c) Se deseja fornecer algum
arquivo para sobrescrever a configuração na tarball modelo
quando o cliente iniciar, consulte
/tftpboot/node_root/drbl_ssi/clients/00_README para mais
detalhes.
[2] Não desejo fornecer serviço Linux diskless ao cliente.
Qual modo prefere ?
[0]

```

No caso, deverá ser escolhida a opção 0 – DRBL SSI, de modo que o servidor armazene as configurações de cada *thin client*. Logo após, é questionado sobre qual modo será utilizado o clonezilla<sup>1</sup>:

```

-----
No sistema, há 4 modos disponíveis para clonezilla:
[0] Modo Clonezilla completo, cada cliente tem seu próprio NFS
baseado em /etc e /var.
[1] Modo Clonezillabox, cada cliente usa tmpfs baseado em /etc
e /var. Neste modo, a carga e o espaço de disco do servidor
necessários é inferior ao Modo Clonezilla completo. Nota! No
modo Clonezillabox, a configuração e os arquivos de
configuração do cliente não serão salvos no servidor DRBL!
Serão usados apenas uma vez e descartados depois que a máquina
desligar!
[2] Não quero clonezilla.
[3] Usar Clonezillalive como o S.O. (Sistema Operacional) dos
clientes (Teste).
Qual modo prefere?
[0]

```

Como o *clonezilla* não será utilizado em conjunto com o DRBL, deve-se escolher a opção 2.

- 5) A etapa seguinte pergunta se o usuário pretende utilizar o disco rígido do *thin client* para criação de memória de troca:

```

-----
Se há um disco local com uma partição swap ou um sistema de
arquivos com permissão de escrita em sua máquina cliente,
deseja usar essa partição swap ou criar um arquivo no sistema
de arquivos com permissão de escrita para que os clientes
tenham mais memória disponível ? (Este passo NÃO destruirá
nenhum dado do disco rígido)
[Y/n]

```

Como os *thin clients* não possuirão discos rígidos, deve-se escolher a opção n.

- 6) As duas etapas seguintes são relativas ao modo como os *thin clients* iniciarão (gráfico ou modo texto) e como será realizado o *login*:

```
-----
Qual modo quer que os clientes usem após a inicialização ?
"1": Modo gráfico (X window system) (padrão),
"2": Modo texto.
[1]
```

```
-----
Qual modo deseja quando o cliente inicie em modo gráfico ?
0: login normal, 1: login automático, 2: login por tempo
[0]
```

Os *thin clients* utilizarão sempre a interface gráfica, bem como os usuários deverão inserir seu *login* e senha para acesso aos computadores. Logo, deverão ser escolhidas as opções 1 e 2.

- 7) As próximas duas etapas são relacionadas à segurança, especificamente sobre utilização de senha no boot e senha específica do usuário root<sup>1</sup> para os *thin clients*:

```
-----
Deseja configurar as senhas de root dos clientes em vez de usar
a mesma senha de root do servidor ? (Para maior segurança)
[y/N]
```

```
-----
Deseja configurar a senha pxelinux dos clientes para que quando
os clientes iniciarem, a senha deva ser introduzida para
iniciar (Para maior segurança)
[y/N]
```

Como níveis de segurança mais avançados não será o foco do trabalho, deve ser escolhida a opção padrão (N) para cada uma das perguntas.

- 8) As três perguntas seguintes são relacionadas ao *prompt* de inicialização dos clientes:

```
-----
Deseja configurar o prompt de inicialização dos clientes ?
[Y/n]
```

```
Quantos décimos de segundo tem a contagem regressiva do prompt
de inicialização dos clientes ?
[70]
```

```
-----
Deseja usar fundo gráfico para o menu PXE quando o cliente
iniciar?
Nota! Se usar o menu gráfico PXELinux, e o cliente falhar ao
iniciar, pode mudar para o modo texto executando
"/opt/drbl/sbin/switch-pxe-bg-mode -m text".
```

[Y/n]

Como os *thin clients* serão computadores antigos reaproveitados, pode ocorrer da placa de vídeo não dar o devido suporte para *menu* gráfico do *prompt*. Assim, é uma boa recomendação escolher a opção Y para a primeira pergunta, manter o padrão (70) para a segunda e, por fim, a opção N para a terceira pergunta.

- 9) A seguir, é questionado se deseja que todos os usuários tenham acesso aos dispositivos:

```
-----
Deseja deixar o áudio, cdrom, disquete, video e plugdev (como
um dispositivo USB) aberto a todos os usuários no cliente
DRBL ? Se sim, serão adicionados todos os usuários aos grupos
de dispositivos no servidor e nos cliente.
[Y/n]
```

Como os usuários terão acesso aos dispositivos da estação, basta teclar *Enter* para confirmar a opção.

- 10)O passo seguinte é para configuração da interface de rede de modo que a mesma possa ter dois IPs distintos:

```
Usando um alias de interfaz, cada cliente pode ter 2IPs,
um deles é um IP privado para os clientes conectados ao
servidor DRBL, e o outro é um IP público para clientes
directamente conectados à WAN pelo switch!
Deseja configurar IPs públicos para os clientes ?
[y/N]
```

Como o servidor será configurado, mais adiante, a funcionar como *Network Address Translation* (NAT), não se faz necessária a criação de alias, bastando teclar *Enter* para escolher a opção padrão.

- 11)A seguir, é perguntado se deseja que os clientes do DRBL possam acessar a estação em modo terminal:

```
-----
Deseja que os clientes DRBL tenham a opção de executar em modo
terminal ?por ex. quer deixar este cliente executar a
visualização remota (que utilizará recursos do servidor),
introduza "Y" aqui.
Nota!
0. Se reponder sim a esta opção, o cliente terá um ambiente
limitado, por ex. NÃO terá acesso local ao USB, CD, audio,
impressora, etc. no cliente.
1. Se seu servidor não é potente, diga "não" aqui.
2. Introduzindo "yes" (sim) aqui, será ativado o xdmcp,
Não é seguro ativá-lo. Configurando /etc/hosts.allow e
/etc/hosts.deny para somente permitir acesso local é outra
alternativa mas não a mais segura.
Usando um firewall para a porta 177 é o mais seguro se deseja
```

```

ter xdmcp ativado.
Leia o manual para mais advertências sobre a segurança do
XDMCP.
Por favor configure-o por você mesmo!
3. Se inserir "yes" (sim) aqui, deve reiniciar seu gerenciador
de ambiente de área de trabalho (gdm/kdm) depois, lembre-se de
salvar seus dados antes de fechar as aplicações!
Deseja que os clientes DRBL tenham a opção de executar em modo
terminal?
[y/N]

```

Como o foco é o uso por usuários que não possuem conhecimento de informática, o uso será voltado exclusivamente para a interface gráfica. Assim, deverá ser teclado Enter para escolher a opção padrão, não havendo necessidade de execução via terminal.

- 12) Para que os clientes possam acessar a Internet, deve ser teclado Enter para escolher a opção padrão na pergunta a seguir, na qual configurará o servidor para funcionar como NAT:

```

-----
Deseja deixar o servidor DRBL como um servidor NAT ? Se não,
seu cliente DRBL NÃO será capaz de acessar a Internet.
[Y/n]

```

- 13) Será, então, perguntado se deseja manter configurações anteriores dos clientes DRBL, caso existam, e é dado um aviso para que, em caso de alteração no *kernel* de modo que não mais suporte o NFS sob UDP ou TCP, que seja novamente executado o comando **drblpush -i** para evitar falha na execução dos clientes. Por último, dado o aviso que segue:

```

*****
Agora estamos prontos para por os arquivos à disposição do
sistema!
Deseja continuar ?
Atenção! Se continuar, as regras de seu firewall serão
sobrescritas durante a configuração!
As regras originais serão salvas com o nome de iptables.drblsave
no diretório de configuração de sistema (/etc/sysconfig ou
/etc/default).
[Y/n]

```

Após teclar Enter para continuar, as configurações do DRBL serão aplicadas ao servidor e a solução estará pronta para uso.

### 3.5 Análise de desempenho das soluções propostas

Para análise das soluções, foram realizados testes no servidor de modo a avaliar o desempenho com relação ao uso de processamento, de memória e de tráfego de rede. Para os testes, foram utilizadas ferramentas específicas para coleta dos dados em quatro momentos distintos, a saber:

- **Etapa 1** - estado do servidor antes do carregamento de qualquer dos *thin clients*, havendo apenas o *login* em modo gráfico de um usuário administrador;
- **Etapa 2** - estado do servidor após o carregamento do *thin client*;
- **Etapa 3** - estado do servidor após o *thin client* abrir o editor de texto *Libre Office Writer*;
- **Etapa 4** - estado do servidor após o *thin client* abrir o navegador de internet *Firefox*.

As etapas de 2 a 4 serão realizadas em três situações distintas: a primeira, com o Thin Client 01, a segunda com o Thin Client 02 e a terceira com ambos os *thin clients*.

#### 3.5.1 Análise do consumo de memória

Para análise do uso da memória, foi utilizada a ferramenta **free**, disponível por padrão no Ubuntu 12.04 e em diversas outras distribuições. O **free** é uma ferramenta que exibe a quantidade de memória livre e em uso, seja ela memória física ou de troca, bem como os *buffers* usados pelo *kernel*. O comando **free** foi executado no servidor ao final de cada uma das quatro etapas descritas na seção 3.5.

### 3.5.2 Análise do consumo de processamento

O uso do processador será analisado com o aplicativo **vmstat**. O **vmstat** exibe informações sobre os processos, memória, paginação blocos de IO (Input/Output – Entrada/Saída) e, entre outros, a atividade do processador. O comando **vmstat** deve ser executado seguido do tempo de atraso para exibição dos resultados, que, no caso do teste, foi de um segundo, dispensando o uso do comando **sudo**. Caso seja executado sem o valor do tempo de atraso, ele exibirá apenas a média dos resultados a partir do momento em que o boot do servidor foi realizado até a execução do comando. O **vmstat** foi executado imediatamente antes da etapa 2, sendo finalizado após a conclusão da etapa 4. Nele, foi utilizado o campo **id** (abreviação de *idle*), que demonstra o quanto há de disponibilidade de processamento. Porém, para a análise do resultado, foi utilizada a diferença entre o tempo ocioso e o total de processamento (100%), de modo a mostrar o quanto o processador realmente trabalhou.

### 3.5.3 Análise do tráfego de rede

Para verificação do tráfego de rede foi utilizada a ferramenta **iptraf**. O **iptraf** foi criado em 1997 por Gerard Paul Java, e atualmente encontra-se na versão 3.0. Ela é uma ferramenta de monitoramento de rede que intercepta os pacotes na rede, fornecendo uma variedade de informações sobre o tráfego IP percorrido. Com ela, é possível verificar a quantidade de pacotes recebidos e enviados, bem como a quantidade de bytes transmitidos. Para o correto funcionamento da ferramenta, ela deve ser executada em conjunto com o comando **sudo**. Uma vez executada, para a obtenção dos dados que serão utilizados no teste, deve-se escolher a opção *Detailed interface statistics*, seguido da interface de rede a ser utilizada, no caso, a **eth0**, conforme configurações descritas nas seções 3.3 e 3.4. Os dados do **iptraf** foram capturados ao final das etapas de 02 a 04, com a informação dos *bytes* e pacotes enviados pelo servidor e recebidos pelos *thin clients*.

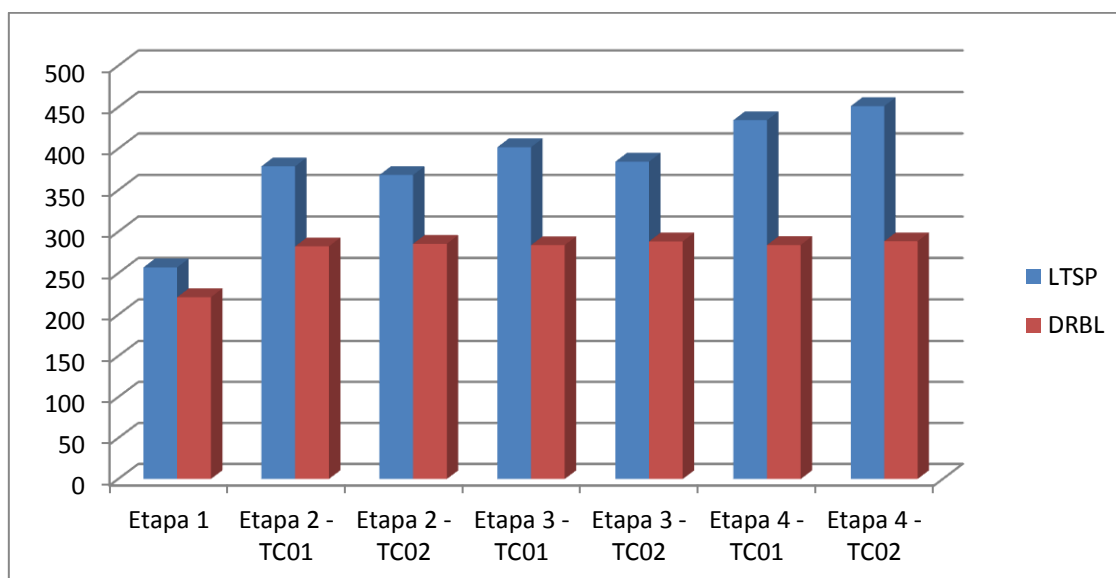
## 3.6 Análise dos dados coletados

Os dados coletados das duas soluções são apresentados em conjunto, de modo a proporcionar uma melhor visualização das vantagens e desvantagens de cada uma das soluções.

### 3.6.1 Memória

Os resultados da coleta de memória nas estações TC01 e TC02 durante as Etapas de 1 a 4 podem ser observados no Gráfico 1.

Gráfico 1 - Evolução do consumo de memória durante as Etapas de 1 a 4



Fonte: próprio autor.

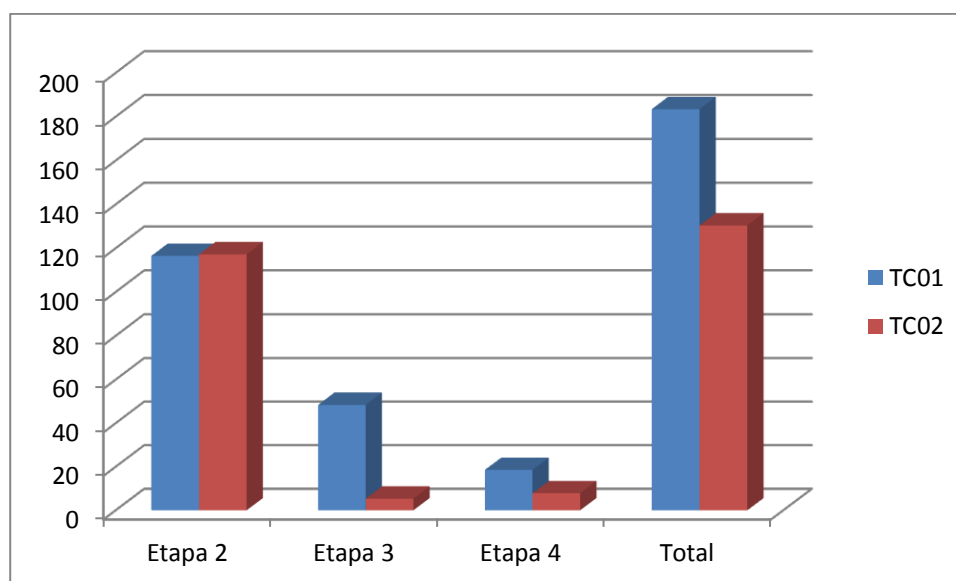
O consumo de memória do LTSP foi crescente, partindo do 257,096MB da Etapa 1 para um consumo médio entre as duas estações de 442,88MB. Ou seja, a partir do estado inicial do servidor até o carregamento do Firefox, houve um consumo adicional de 177,296MB no TC01, e de 194,272MB no TC02.

Já com relação ao DRBL, houve apenas um aumento da memória consumida pelo servidor imediatamente após o carregamento do *thin client*, mantendo consumo

de memória estável nas demais etapas: do consumo inicial de 221,132MB, para um consumo médio de 286,25MB ao final da Etapa 4, havendo um consumo de 62,92MB após o carregamento do TC01 e de 65,118MB quanto ao TC02. Vale ressaltar que o consumo de memória ocorreu apenas na Etapa 2, com variações de consumo pouco significativas nas demais etapas, conforme pode ser observado no Gráfico 1.

O consumo de memória com ambas as estações carregadas juntas não causou alterações significativas no DRBL, mantendo praticamente a mesma média de consumo de uma única estação, ou seja, a quantidade de estações não interfere no consumo de memória RAM do servidor. Já quanto ao LTSP, o consumo de memória não é diretamente proporcional à quantidade de estações, pois com relação às Etapas 3 e 4, não houve um aumento significativo do consumo de memória. O Gráfico 2 mostra o consumo de memória do servidor com o carregamento do TC02 imediatamente após o TC01.

Gráfico 2 - Consumo de memória com carregamento sequencial das estações TC01 e TC02



Fonte: próprio autor.

Ou seja, à exceção da Etapa 2, onde o consumo de memória foi equivalente após o carregamento das duas estações, nas etapas 3 e 4, o consumo de memória da estação TC02 foi drasticamente menor que TC01. A média de consumo do carregamento individual das estações, mais o consumo do TC01 durante o carregamento simultâneo, é de 184,991MB. Já o consumo individual da estação

TC02, após carregamento em conjunto com a estação TC01, foi de 130,358MB, ou seja, 29,53% menor.

Com estes valores, pode-se concluir que o consumo adicional de memória por estação, utilizando-se simultaneamente os aplicativos Libre Office Writer e Firefox, é o demonstrado na Tabela 5.

Tabela 5 - Memória adicional necessária ao servidor

<b>SERVIDOR - MEMÓRIA NECESSÁRIA (POR ESTAÇÃO)</b>		
	<b>1ª estação</b>	<b>Estação adicional</b>
DRBL	62,773MB	0,0MB
LTSP	184,991MB	130,358MB

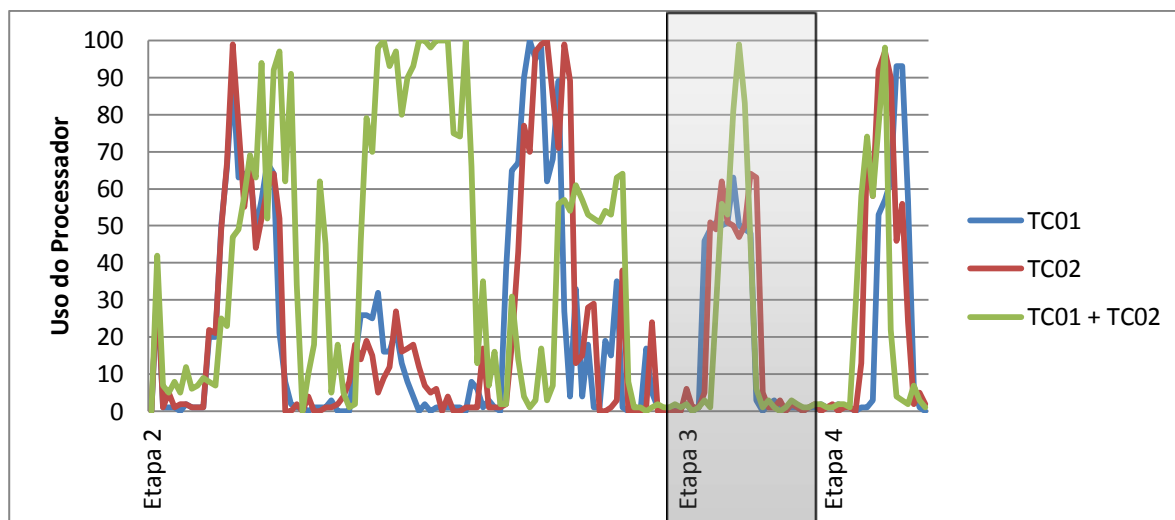
Fonte: próprio autor.

Assim, o servidor de uma rede com oito *thin clients*, utilizando a solução DRBL, necessitará de 284MB de memória RAM, enquanto a mesma rede utilizando o LTSP necessitará de 1,35GB para um funcionamento satisfatório.

### 3.6.2 Processador

O resultado do uso do processador durante o teste utilizando a solução LTSP pode ser visto no Gráfico 3.

Gráfico 3 - Uso do processador no servidor LTSP

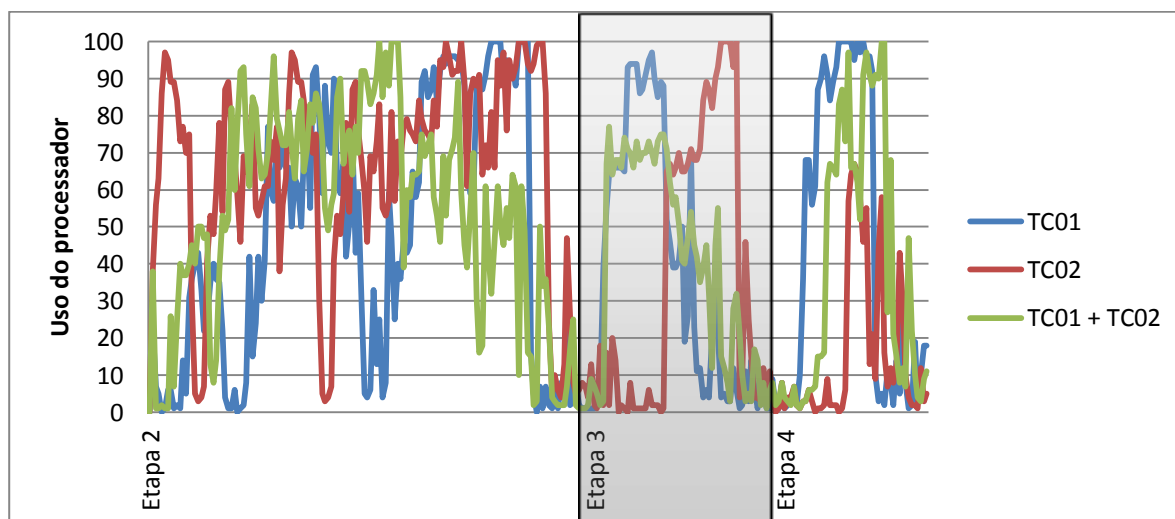


Fonte: próprio autor.

À exceção da Etapa 2, com a utilização simultânea das estações TC01 e TC02, o processador se comportou de forma regular, chegando próximo de 100% apenas durante o carregamento dos aplicativos.

Com relação ao DRBL, o resultado do uso do processador pode ser visto no Gráfico 4.

Gráfico 4 - Uso do processador no servidor DRBL



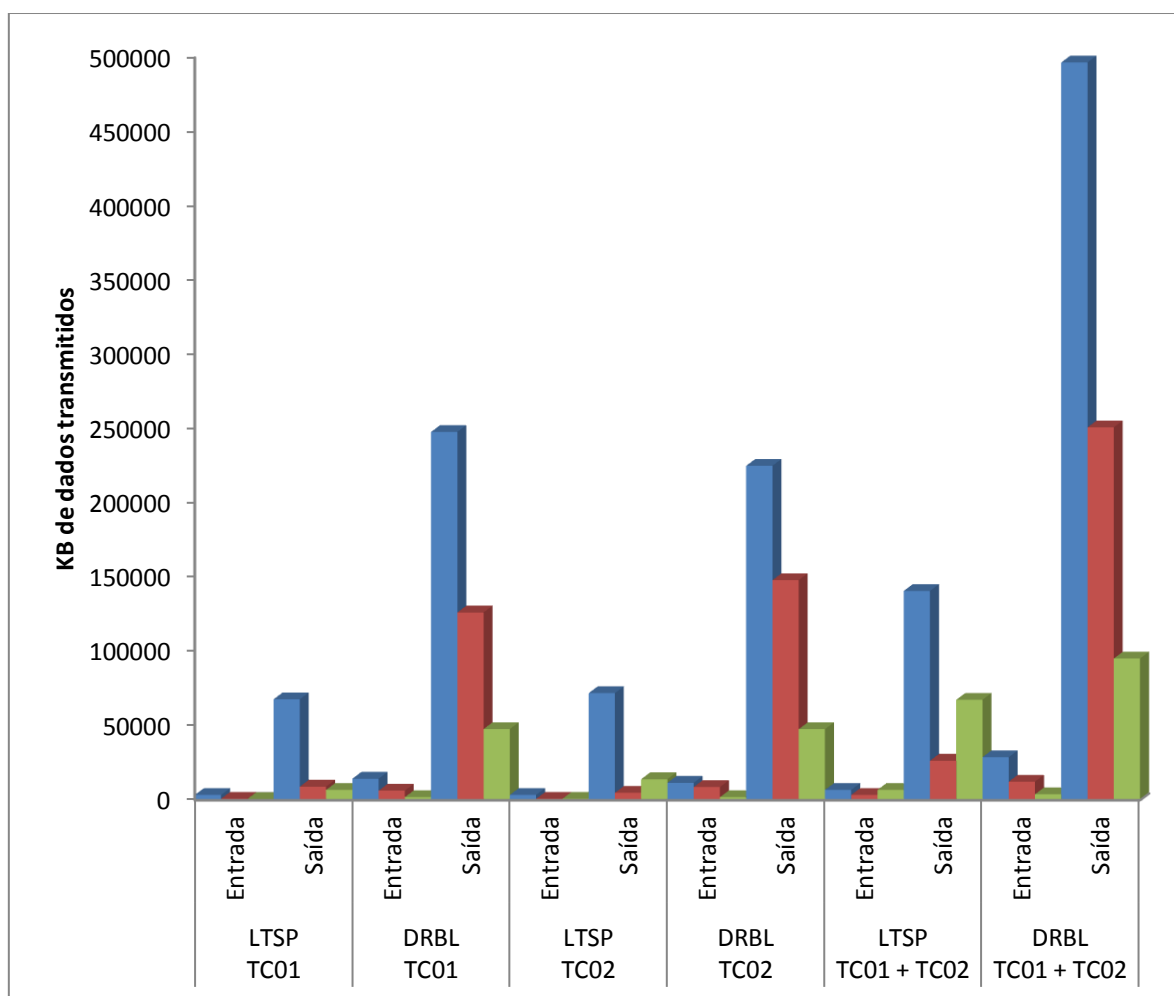
Fonte: próprio autor.

Na solução DRBL, durante a Etapa 2, em quaisquer dos cenários houve um uso elevado do processador, o qual atuou acima dos 50% na maior parte dos testes. Mesmo nas demais etapas, o uso do processador foi bastante intenso.

### 3.6.3 Tráfego de rede

Os dados do tráfego de rede capturados pelo iptraf em cada uma das soluções podem ser observados no Gráfico 5. **Erro! Fonte de referência não encontrada..**

Gráfico 5 - Tráfego de dados

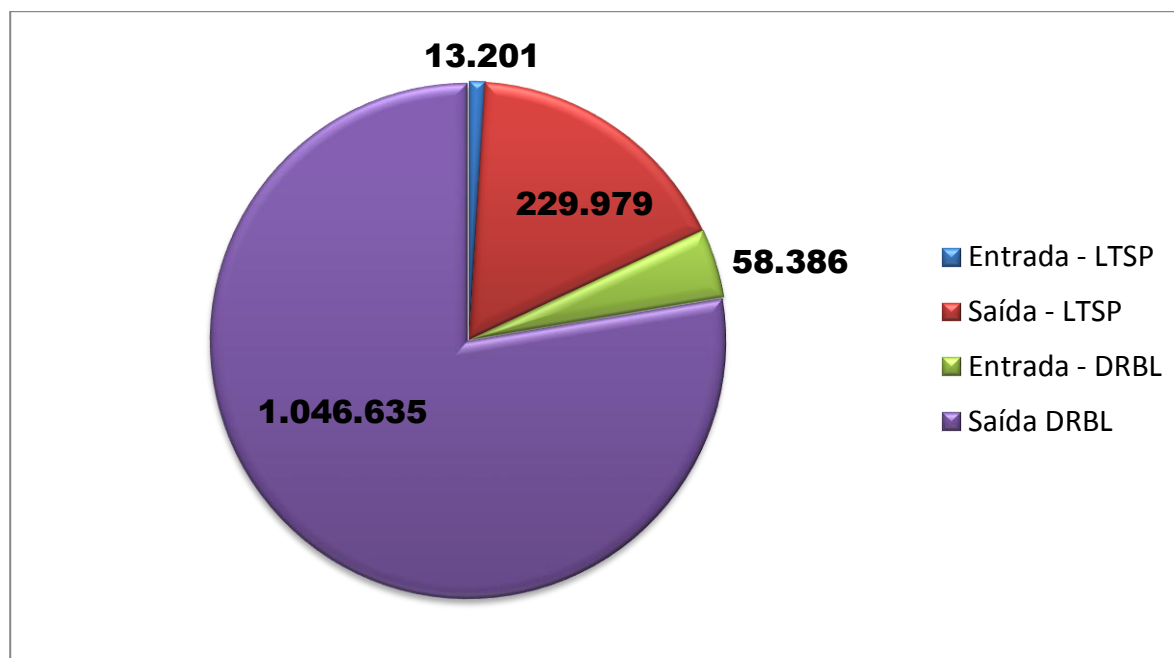


Fonte: próprio autor.

O primeiro ponto a ser observado é a grande diferença entre o total de dados transmitidos pelo LTSP em relação ao DRBL. Este último, devido à necessidade da transferência dos dados do servidor para que sejam processados nas estações, acaba por resultar em um tráfego muito maior em comparação ao LTSP, já que este, com o uso do XDMCP, transmite basicamente as instruções do que deve ser

desenhado pela aplicação do Servidor X, carregada nas estações. Essa diferença pode ser melhor observada no Gráfico 6.

Gráfico 6 - Total de KB transmitidos durante testes da estação TC01



Fonte: próprio autor.

O segundo ponto a ser observado é que, como as estações apenas solicitam as informações a ser fornecida pelo servidor, a entrada de dados é muito menor do que a saída.

### 3.7 Solução escolhida

À exceção dos testes de consumo de memória, a solução LTSP teve um consumo significativamente menor de recursos que a solução DRBL. Levando-se em consideração que memória RAM causa pouco impacto no custo final da implementação de rede devido ao seu baixo custo, este critério acaba por ter menor importância na escolha. Além do mais, o DRBL exige que o hardware dos *thin clients* seja capaz de suportar o sistema operacional: analisando-se a Tabela 6, a qual apresenta os requisitos mínimos e recomendados para a instalação, pode-se

concluir que as exigências de hardware para funcionamento do sistema operacional vão em sentido oposto a um dos objetivos deste trabalho, que é o de reaproveitamento de hardware antigo.

Tabela 6 - Requerimentos de hardware para instalação da versão Desktop do Ubuntu 12.04

<b>REQUISITOS MÍNIMO E RECOMENDADO</b>		
<b>Componente</b>	<b>Mínimo</b>	<b>Recomendado</b>
Processador	Pentium 4 1Ghz	Pentium 4 1.5Ghz
Memória RAM	512MB	1024MB
Disco rígido	5GB	10GB

Fonte: Novais, 2012.

Mesmo que se pretenda adequar o hardware das estações, os custos para tal pretensão ultrapassariam em muito os mesmos custos para adequação do servidor, utilizando-se a solução LTSP.

Todos estes fatores levaram à escolha do LTSP como solução mais adequada para este trabalho.

## 4 RESULTADOS

Este capítulo apresenta os resultados da montagem do laboratório de informática para demonstrar a viabilidade da solução escolhida. Após a análise e escolha da solução LTSP, apresentada no capítulo 3, apresenta-se aqui qual o ambiente utilizado (Seção 4.1), sua configuração inicial (Seção 4.2), a implementação propriamente dita (Seção 4.3), a validação (Seção 4.4) e as conclusões obtidas (Seção 4.5).

### 4.1 O ambiente utilizado

Para a implementação da solução *utilizando-se o LTSP* foi necessário a utilização de um servidor previamente configurado que dispunha das seguintes configurações descritas na Tabela 7.

Tabela 7 - Configurações do Servidor LTSP

<b>SERVIDOR LTSP</b>	
Processador:	AMD Athlon X6 2,4Ghz
Memória:	8GB DDR3
Disco rígido:	512GB
Interface de rede	Duas placas de rede Realtek RTL8139

Fonte: próprio autor.

O local já possuía uma conexão com Internet de banda larga de 2mb/s, sendo necessário apenas montar uma rede de 100mbps apenas para a utilização da solução, utilizando um roteador e dois switches que conectavam os computadores através de cabos UTP, podendo-se transferir dados a uma taxa máxima de até 12,4MB/s.

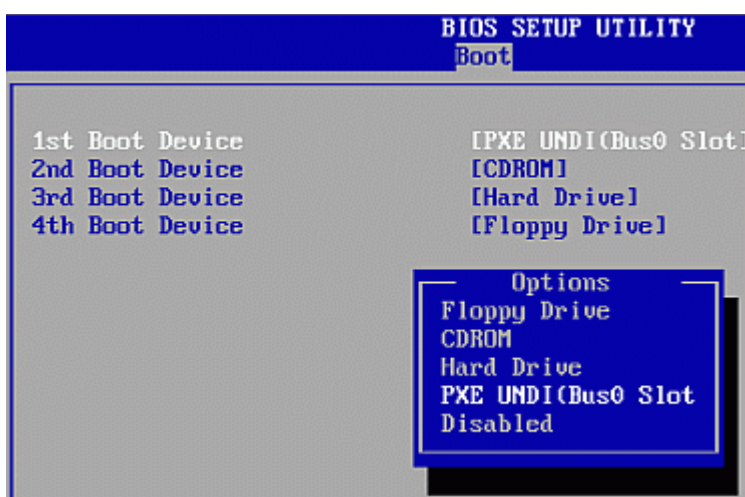
Os computadores utilizados possuíam configurações totalmente distintas, utilizando-se desde computadores antigos, objeto da proposta do trabalho, bem como notebooks, configurando-os para utilizar todos os recursos, serviços e aplicações disponibilizadas pelo servidor, sem a necessidade de utilizar um sistema

operacional para cada computador.

## 4.2 Configurações Iniciais

No processo de configuração inicial foi verificado se todos os computadores utilizados na solução dispunham de boot pela placa de rede (PXE), o que foi verificado apenas nos notebook e computadores mais recentes. Nas máquinas que não possuíam este recurso, foi apenas realizado a configuração na BIOS, definindo o PXE como boot inicial, conforme mostra a Figura 20.

Figura 20 - Ajuste da BIOS para boot via PXE

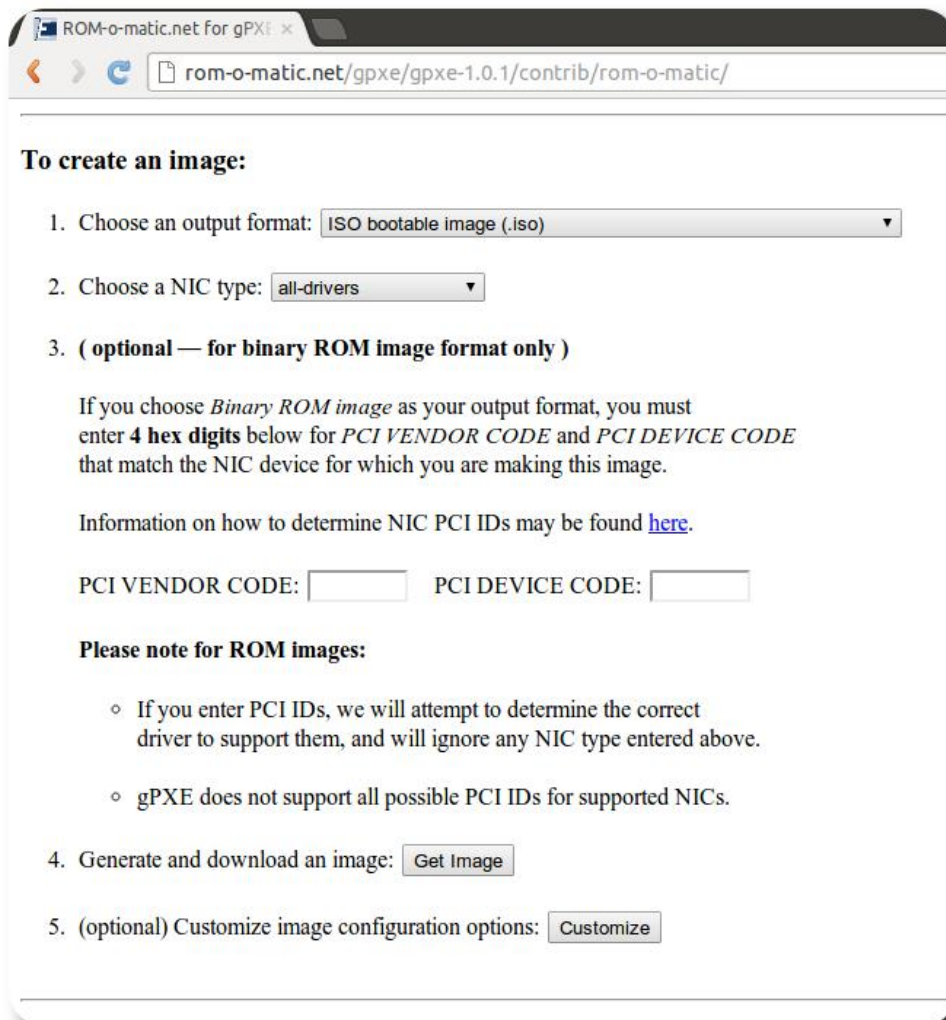


Fonte: K5, 2012.

Essa configuração é necessária para que os computadores da rede possam realizar a inicialização do sistema operacional e as demais aplicações disponibilizadas pelo servidor. Já para os computadores que não possuíam este recurso, foi necessário criar um CD de boot contendo o gPXE, fornecido pelo ROM-o-matic.net (GPXE, 2012). Conforme visto na Figura 21, no site, é possível acessar o *image generator*, escolher o dispositivo para o qual será gerado a imagem (disquete, CD, pendrive, etc) e o tipo de placa de rede. É possível gerar-se uma imagem contendo todos os tipos fornecidos pelo site, o que, mesmo assim, cria uma imagem com apenas 694 kilobytes, o que a torna a escolha recomendada, já que cabe em qualquer das mídias disponíveis. Com a imagem gerada, bastou então gravá-la em

CDs e utilizá-los nas máquinas onde o PXE não estava disponível.

Figura 21 - Gerador de imagens do site www.rom-o-matic.net



The screenshot shows a web browser window with the address bar displaying "rom-o-matic.net/gpxe/gpxe-1.0.1/contrib/rom-o-matic/". The page content is as follows:

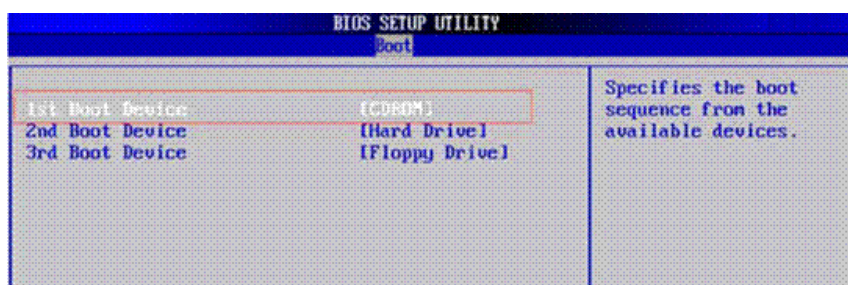
**To create an image:**

1. Choose an output format:
2. Choose a NIC type:
3. ( optional — for binary ROM image format only )  
If you choose *Binary ROM image* as your output format, you must enter **4 hex digits** below for *PCI VENDOR CODE* and *PCI DEVICE CODE* that match the NIC device for which you are making this image.  
Information on how to determine NIC PCI IDs may be found [here](#).  
PCI VENDOR CODE:  PCI DEVICE CODE:
4. Generate and download an image:
5. (optional) Customize image configuration options:

Fonte: GPXE, 2012.

O uso do CD de boot é bastante simples, bastando apenas realizar uma configuração na BIOS, alterando novamente a ordem de boot, porém, deixando desta vez o CD-ROM como boot inicial, conforme Figura 22.

Figura 22 - Ajuste da BIOS para boot via CDROM



Fonte: WINDOWS SERVER PASSWORD RECOVERY, 2012.

Desta forma, uma vez ligado o computador, inicialização será feita através do CD de boot, que, por sua vez, irá identificar o driver correto da placa de rede, carregando o gPXE para iniciar a negociação com o servidor DHCP, que irá atribuir um endereço IP e as demais configurações da rede para, assim, realizar a transferência de arquivos necessários para a inicialização do sistema operacional de cada *thin client*.

Após as configurações realizadas nas etapas anteriores, os computadores foram ligados em rede e iniciados seus respectivos processos de boot, inicializando aos computadores normalmente deixando-os na tela de *login*, conforme mostra a Figura 23.

Figura 23 - Exibição da tela de *login* das estações *thin client*

Fonte: próprio autor

### 4.3 Implementação

A implementação foi realizada em um laboratório composto por oito computadores e um servidor com as características já descritas na seção 4.1. Alguns dos computadores utilizados na implementação eram máquinas dispensadas por já serem ultrapassadas em termos de hardware, mas para a implementação proposta neste trabalho, o uso delas foi considerado essencial, pois pôde-se utilizá-las e testá-las por inúmeras vezes, constatando que seu uso não depende simplesmente de estar atualizado em termos de tecnologia, mas sim da necessidade. Ficou evidente que estes equipamentos, considerados “sucata” podem muito bem ser utilizado como uma forma de inclusão digital.

Dentre os computadores utilizados, havia configurações que variavam desde um Pentium III com 500 MHz a um dual core i3 com 2.266 MHz.

Na Tabela 8 é possível notar as diferenças encontradas nas configurações dos computadores utilizados. Essas diferenças não impactaram diretamente na implementação, mas ocorreram problemas com relação aos hardwares mais antigos, principalmente no quesito manutenção: como alguns dos computadores sequer ligaram ou tinham componentes que não funcionaram adequadamente, foi necessário realizar a troca de peças, resultando em perda de tempo na desmontagem e montagem dos mesmos. Por exemplo, nos computadores onde foi necessário dar boot via CD, os leitores não reconheciam a mídia, sendo necessário trocar pelo leitor de equipamentos que possuíam boot via PXE. Em um dos computadores, foi necessário trocar a placa-mãe, já que o mesmo não iniciava. No entanto, uma vez que estes problemas haviam sido sanados a utilização do LTSP e os testes ocorreram de forma muito direta e tranquila.

Tabela 8 - Configuração de hardware dos *thin clients*.

<b>Máquina 1</b>	<b>Máquina 2</b>
Pentium III 500 mhz Socket 370	AMD E-350 1.6 ghz 2 núcleos
128MB DIMM PC100	2GB DDR3 (648 compartilhada com vídeo)
Vídeo SIS630	Vídeo Radeon HD 6310
Placa SIS900	Rede Realtek RTL8111
<b>Máquina 3</b>	<b>Máquina 4</b>
Pentium IV 1Ghz Socket 478	Pentium Dual Core CPU E220 2.4Ghz
256MB DDR2	2GB DDR2
Vídeo VIA Pro Savage 8	Vídeo Intel 945G
Rede via 6103	Rede RTL8161

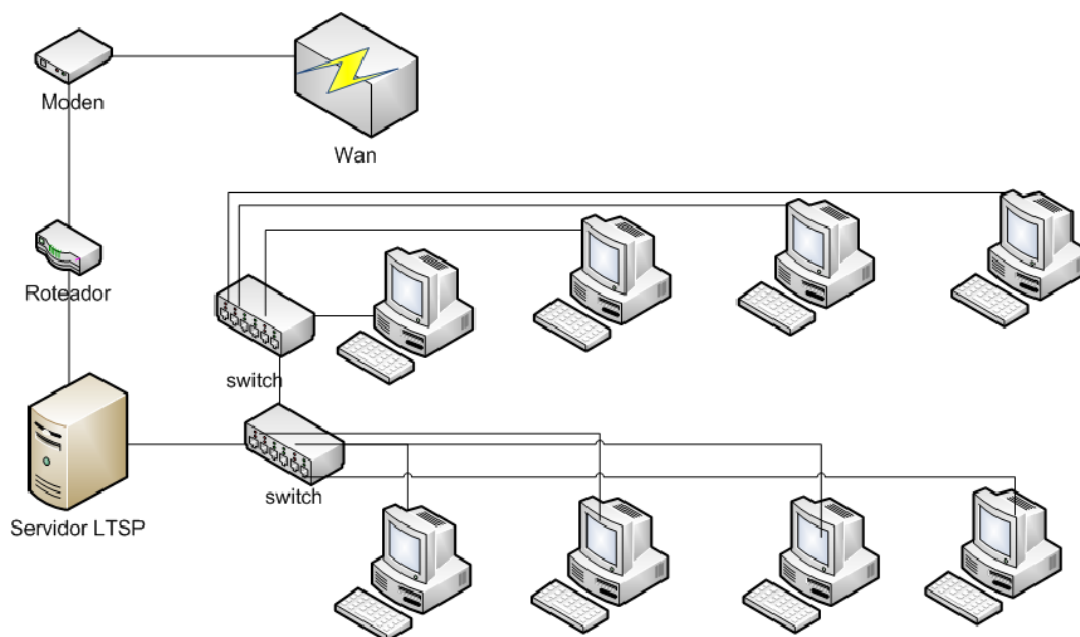
<b>Máquina 5</b>	<b>Máquina 6</b>
Pentium III 600MHz	Dual core i3 350m 2266 mhz
256MB DIMM	2GB DDR3
Nvidia TNT2	Intel HD Graphics Ironlake-M
Via Rhine VT 6105	Rede Realtek PCI-E FE
<b>Máquina 7</b>	<b>Máquina 8</b>
Atom N270 1,6Ghz	Atom N280 1,666Ghz
1GB DDR2	2GB DDR2
Intel 945	Intel 945
Realtek RTL 8102e	Realtek RTL8102e

Fonte: próprio autor

Os softwares escolhidos para implementar a solução foram todos baseados em soluções livres, como o Libre Office, Firefox, Gimp, entre outros considerados necessários para o uso no dia-a-dia de usuários que estão começando a utilizar o computador. Este era um dos principais objetivos do trabalho: gerar um baixo custo de implementação. Neste caso isso foi atingido pela utilização de softwares que não possuem custo de licenciamento e também pela doação de equipamentos antigos.

O custo efetivo com hardware foi apenas com o servidor e equipamentos de rede (cabos e switches) utilizados. Esses custos foram necessários para a implementação do cenário exibido na Figura 24.

Figura 24 - Mapa da rede.



Fonte: próprio autor.

O local escolhido já possuía acesso a internet com uma velocidade de uso residencial de 2 megabits, fornecida através de um Access Point no qual o servidor se conectava através de uma placa de rede sem fio. Como o switch utilizado possuía apenas oito portas, foi necessário a utilização de um segundo switch para conexão do servidor, que é o único ponto de acesso à Internet para os computadores dentro da rede.

O servidor, conforme descrito na Seção 3.3, foi configurado para atribuir dinamicamente os IP necessários para cada computador, bem como suas configurações necessárias. Além do sistema operacional, provedor de internet e demais aplicações, o servidor concentra quase todo o custo necessário, sendo o único equipamento onde todas as peças de hardware são exigidas, já que os *thin clientes* não possuem disco-rígido, trabalhando apenas para disponibilizar recursos de entrada e saída do usuário, sendo as informações processadas apenas no servidor, deixando, assim, os *thin clientes* livres, razão pela qual não necessitam de hardware potente.

O segundo Morimoto (2006), não há uma especificação de servidor correta para uma quantidade de computadores, o que é necessário é fazer um análise do desempenho do servidor. Esta análise foi realizada no Capítulo 3 deste trabalho, e a partir dela, pode-se concluir pelas especificações do servidor que o mesmo seria capaz de suprir mais de 50 *thin clients*. Entretanto, uma quantidade grande de estações fazia com que aumentassem as necessidades de largura de banda, sendo necessário a utilização de redes de 1 gbps ou até mesmo a utilização de fibra óptica, o que encareceria a implementação da rede.

O cabeamento pôde ser considerado a parte mais simples da implementação. Depois que os computadores foram colocados na bancada, não houve problemas em passar os cabos por se tratar de uma rede pequena, além dos computadores estarem todos no mesmo setor e na mesma bancada. Para evitar problemas de conexão, os cabos precisam estar firmemente conectados. É aconselhável também evitar que os switches se movimentem posicionando-os longe do contato com os usuários, já que uma falha de comunicação faria com que os *thin clients* deixassem de coletar dados do servidor, passando então a não mais funcionar adequadamente, tendo como resultado a tela congelada até que a conexão com o servidor fosse restabelecida.

## 4.4 Validação

Após instalada a rede, o passo seguinte foi realizar os testes necessários para validar se a solução era viável ou não. O primeiro item que pôde ser ressaltado foi a rapidez na inicialização dos *thin clients*, onde todos os oito computadores iniciaram de forma bastante rápida em comparação a outros sistemas operacionais ou mesmo *tick clients*. O carregamento dos aplicativos, bem como o desempenho no uso na Internet também obteve bons resultados.

Porém, a solução não se mostrou viável com relação à exibição de vídeos, sejam eles através da Internet ou em um arquivo armazenado no servidor. Bastaram dois computadores reproduzindo vídeos para que o consumo da banda de rede atingisse o seu limite de 12,4MB, o que fez com que os demais computadores se tornassem menos responsivos, com atraso nas atualizações de tela ou respostas do teclado. Esse problema ocorre porque, como transmissão do vídeo precisa ser processada pelo servidor e enviado quadro a quadro para os computadores via XDMCP, considerando-se que os vídeos estavam sendo reproduzidos a 30 quadros por segundo, isso fez com que o servidor enviasse as atualizações de tela com capturas de sessenta imagens por segundo, o que é inviável em uma rede de 100 megabits.

Para as demais aplicações, como Open Office, Gimp ou Firefox, o uso em conjunto das mesmas por todos os computadores foi satisfatório, sendo que todos funcionaram perfeitamente, sem agredir o nível de processamento do servidor em nenhum momento e sem causar travamentos ou lentidão na rede, mantendo sempre o tráfego da rede nos 40% de uso.

Um dos motivos para o baixo consumo de processamento e memória (levando-se em consideração que oito máquinas estavam utilizando recursos exclusivamente do servidor), inclusive diferenciando a solução escolhida das demais, é quanto ao compartilhamento de recursos já processados pelo servidor entre as demais estações. Por exemplo, ao abrir o Firefox, o servidor carrega-o na memória e disponibiliza para o computador solicitante. Caso outro computador solicite a mesma aplicação, o servidor não irá carregar na memória outro processo do Firefox; irá apenas repassarão novo solicitante o que já se encontra na memória,

atendendo de forma mais rápida o segundo solicitante do que foi quanto ao primeiro, ou seja, qualquer aplicação que já se encontre aberta na memória, não será carregada em duplicidade, melhorando assim o desempenho do servidor e atendendo os solicitantes de modo mais rápido e eficiente. Em outras palavras, o uso de reentrância de memória e a criação de um novo bloco de dados quando outra estação inicia o mesmo processo gera uma tremenda economia de memória no servidor.

Todos os testes foram realizados com todos os computadores ligados a rede, visando sempre a analisar a funcionalidade da solução simulando um ambiente de uso constante, de forma a buscar os possíveis problemas que poderiam vir a existir no decorrer da implementação, sendo que a implementação e os testes levaram cerca de 8 horas, abrindo e fechando programas, ligando e desligando os computadores para verificar inclusive possíveis mudanças nos resultados coletados.

Todas as etapas efetuadas até a finalização da implementação foram acompanhadas de perto, analisando cada erro e cada acerto, para se chegar às conclusões finais descritas na seção 4.5.

## **4.5 Conclusões das Validações**

Após todas as validações concluiu-se que a solução proposta tem um custo baixíssimo, envolvendo apenas mão de obra barata, pouca manutenção, sem custo de licenciamento de software e com custo de hardware baixíssimo.

Podem existir certa resistência por parte dos usuários, por tratar-se de um sistema operacional pouco utilizado como desktop. No entanto, acredita-se que os conceitos aprendidos neste ambiente são facilmente transferíveis para outros sistemas operacionais e aplicativos mais populares. Ao mesmo tempo, os autores notam uma certa resistência por parte do uso de sistemas livres no Brasil, especialmente tratando-se do sistema operacional.

Uma limitação importante do LTSP foi a impossibilidade de se assistir vídeos de forma satisfatória em diversas estações. O gargalo, neste caso, ocorreu justamente na rede, que é fundamental para todos os computadores, ou seja, se o

servidor parar ou a comunicação com este não for possível todos os demais computadores também deixarão de funcionar.

Para concluir, os resultados mostraram que se a reutilização de computadores antigos for considerada a sério, a maciça adoção da solução aqui proposta, além de contribuir para a preservação do meio ambiente, pode auxiliar enormemente a inclusão digital. Ou seja, a solução aqui proposta, definida e analisada é válida e possui excelente retorno. Investir nela é uma ótima forma de se promover a inclusão digital e melhorar o meio ambiente.

## 5 CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho apresentou um estudo de caso sobre a viabilidade da montagem de um laboratório de informática de baixo custo, tanto relacionado ao software quanto ao hardware, através do uso de um sistema operacional e demais aplicativos de uso livre, e do reaproveitamento de hardware considerado obsoleto. Tudo isso foi motivado pela necessidade de combate à exclusão digital, pois praticamente um terço da população brasileira encontra-se nesta situação.

A solução implementada utilizou o conceito de *thin clients*, com o uso da solução LTSP. Os resultados mostraram que a rede mostrou-se funcional, com bom desempenho para o que se propõe, além de que, por possuir um gerenciamento centralizado no servidor e uma distribuição de Linux de fácil usabilidade, foi perfeitamente possível gerenciá-la em tarefas cotidianas, sem a necessidade de conhecimentos avançados em informática.

Além disso, devido sua estrutura, na qual os IPs são atribuídos dinamicamente, sem a necessidade de ajustes no servidor, a expansão da rede se tornou algo trivial. Caso se queira adicionar novas estações, bastará adicionar os novos *thin clients* à rede, ajustando-os para que iniciem via PXE ou através do CD-ROM, que pode ser replicado, já que contém *drivers* para diversas interfaces de rede. Quanto aos requisitos de processamento, a necessidade de velocidade do processador é algo difícil de quantificar, já que depende muito do quão intenso será o uso da rede, bem como quais aplicativos serão utilizados e com que frequência. Por isso, apesar dos testes de validação aqui realizados, torna-se necessário que o responsável por administrar a rede realize testes adicionais para avaliar se o desempenho ainda é satisfatório mesmo após a inclusão de mais estações.

Uma questão importante a ser observada é com relação ao reaproveitamento de hardware. Como as estações são computadores obsoletos, muitas vezes as mesmas dependem de alguma manutenção para o devido funcionamento, porém, isso muitas vezes acaba por ser algo dificultoso, seja pelo tempo despendido para sua realização, seja pela dificuldade em se encontrar alguma peça de reposição necessária

Outra limitação observada foi a questão multimídia. Devido à rede suportar apenas 100Mbit/s, a exibição de vídeo se tornou inviável. De oito computadores em

rede, apenas um conseguiu assistir a um vídeo de forma satisfatória por vez. Uma segunda estação também conseguiria assistir, simultaneamente e de forma satisfatória, mas isso implicaria no consumo máximo do tráfego de rede, trazendo, como efeito colateral, a indisponibilidade das demais estações, ocorrendo demora nos pedidos de entrada e saída de dados. Assim, caso a rede queira ser voltada também para este fim, deve-se buscar a utilização de redes com maior largura de banda, embora isso possa gerar custos maiores na implementação. Por exemplo, a implementação da rede com a utilização de fibra óptica conseguiria aumentar a largura de banda na ordem de 200Mbits a 10gigabits por segundo. Porém o dispêndio de dinheiro com placas de rede apropriadas, bem como da própria fibra óptica, fariam com que a implementação deixasse de ser considerada de baixo custo.

Entretanto, a rede se mostrou funcional para uso de tarefas cotidianas, como a edição de textos e o acesso à Internet, além de seu custo ter sido voltado apenas ao servidor, que possui características próximas às de computadores de uso comum, e com cabeamento e switch, equipamentos de baixíssimo custo. Assim, pode-se concluir que a rede é viável para cumprir o objetivo proposto, qual seja o combate à exclusão digital.

## **5.1 Trabalhos futuros**

Mantendo o foco na inclusão digital, uma questão de suma importância não abordada neste trabalho é quanto à criação de uma metodologia de ensino e elaboração de material apropriado.

A utilização de rede de computadores para uso em centros públicos de acesso mostrou-se uma ferramenta eficiente para o combate à exclusão digital, como ficou demonstrado capítulo de introdução deste trabalho. Entretanto, a disponibilização da rede como ferramenta é apenas uma parte do necessário para a inclusão digital. Faz-se necessário a elaboração de uma metodologia de ensino, a ser utilizada pelos gestores da rede durante o ensino e acompanhamento dos usuários que a utilizam. Principalmente pelo fato de a rede utilizar um sistema

operacional e aplicativos que não são os mais utilizados no mercado, é importante que o usuário, durante o aprendizado, adquira um conhecimento mais geral das ferramentas, já que ensiná-lo a utilizar um editor de texto e uma planilha de cálculo está apresentando conceitos mais abrangentes do que um ensino focado no Word e Excel, por exemplo.

Uma vez concluída a metodologia de ensino, deve-se partir para a elaboração do material propriamente dito, seja através de apostila, seja através de atividades a serem apresentadas via projetor. O material deve ser focado no público-alvo, já que a realidade e necessidades dos usuários de uma rede instalada em um abrigo de menores é totalmente diferente dos usuários de uma da rede instalada em um centro de capacitação de trabalhadores.

## REFERÊNCIAS BIBLIOGRÁFICAS

ANSARI, N.; TIWARI, S.; AGRAWAL, N. **Practical handbook of thin-client implementation**. Nova Deli, New Age International (P) Ltd. Publishers, 2005, 214p.

BELLO, Moises Rocha; MACEDO, Zenon Frota. **Proposta de Arquitetura e Modelo de Gestão do Laboratório de Informática para a Escola Classe 02 / Paranoá-DF: Um Projeto de Inclusão**. 2008. 80 f. Trabalho de conclusão de curso (Licenciatura) – Departamento de Ciência da Computação, Universidade de Brasília, Brasília, 2008.

BEZERRA, Artur Lopes. **Thin Clients com LTSP – Linux Terminal Server Project Uma Alternativa de Sucesso em Ambientes Corporativos**. 2009. 91 f. Trabalho de conclusão de curso (Graduação) - Departamento de Ciência da Computação, Faculdade Lourenço Filho, Fortaleza, 2009.

CENTRO DE POLITICA SOCIAIS. **Mapa da Inclusão Digital**. Disponível em: < <http://www.cps.fgv.br/cps/telefonica/> >. Acesso em: 04 set. 2012.

DAVIS, Euan, PARKER, Andrew, GRAY, Benjamin, KING Onica. **Green Benefits Put Thin-Client Computing Back On The Desktop Hardware Agenda**. Forrester, 2008. Disponível em < <http://www.vxl.co.uk/fckeditor/editor/filemanager/connectors/aspx/fckeditor/userfiles/file/VXL%20Green%20Benefits%20Thin%20Clients.pdf>>. Acesso em: 07 set. 2012.

DRBL. **About DRBL**. Disponível em: < <http://drbl.sourceforge.net/> >. Acesso em: 09 out. 2012.

DRBL. *What are the differences between DRBL and LTSP*. Disponível em: <[http://drbl.sourceforge.net/faq/fine-print.php?path=./1\\_Common/01\\_DRBL\\_LTSP.faq](http://drbl.sourceforge.net/faq/fine-print.php?path=./1_Common/01_DRBL_LTSP.faq)>. Acesso em: 09 out. 2012.

ELETRÔNICA, TMG. **Circuito integrado em DIP**. Disponível em: < <http://www.tmgeletronica.com.br/SubCategoria-Circuito-Integrado-em-DIP-79.aspx>>. Acesso em: 17 set. 2012.

ETHERBOOT PROJECT. **Etherboot/gPXE Wiki**. Disponível em: <<http://etherboot.org/wiki/start>>. Acesso em: 09 out. 2012.

FARIA, Clístenes Pereira de Sousa. **Thin Clients: Soluções, Implementação e Desempenhos**. 2009. 40 f. Trabalho de conclusão de curso (Graduação) - Departamento de Informática e Estatística, Universidade Federal do Piauí, Piauí, 2009.

FERNANDES, Natalia Castro. **O que é SSH**. Disponível em: <[http://www.gta.ufrj.br/~natalia/SSH/o\\_q\\_e\\_ssh1.html](http://www.gta.ufrj.br/~natalia/SSH/o_q_e_ssh1.html)>. Acesso em: 12 nov. 2012.

GAVIDIA, Jorge Juan Zavaleta. **TFTP ( Trivial File Transfer Protocol )**. Disponível em:< <http://penta2.ufrgs.br/rc952/trab1/tftp.html>>. Acesso em: 18 out. 2012.

GPXE. **ROM-o-matic.net**. Disponível em: [www.rom-o-matic.net](http://www.rom-o-matic.net)>. Acesso em 20 out. 2012.

GREENBERG, S. (2000). **What are the Real Benefits?** 2000. Disponível em: <[http://www.thinclient.net/technology/technology\\_review.htm](http://www.thinclient.net/technology/technology_review.htm)>. Acesso em: 28 ago. 2012.

K5, Computer Lab. **Deploy XP Images in the Computer Lab With Windows Deployment Services on Server**. Disponível em: <<http://oakdome.com/k5/tutorials/deploy-xp-images/deploy-xp-with-windows-2003-deployment-services-2.php>>. Acesso em: 03 nov. 2012.

MICHELON, Gisane Aparecida; HILD, Giancarlo Fernando. Reutilização de Computadores Obsoletos com a Implementação de um Servidor de Terminais GNU/Linux. *Informática Pública*, ano 11 (1), p. 89 – 105, 2009.

MORIMOTO, C.E. **Redes Guia completo**. 3ª Edição. 2003. e-books Disponível em: <[www.guiadohardware.net](http://www.guiadohardware.net)>. Acesso em: 18 ago. 2012.

MORIMOTO, C.E. *Redes e servidores Linux: guia pratico*. 2ª.ed.Porto Alegre: Sul Editores, 2006, 448 p.

NOVAIS, Claudio. **Como instalar o Ubuntu 12.04 Precise Pangolin (e primeiros passos)**. Disponível em: < <http://ubuntued.info/como-instalar-o-ubuntu-12-04-precise-pangolin-e-primeiros-passos>>. Acesso em 12 out. 2012.

OI FUTURO. **Oi conecta**. Disponível em:< <http://oiconecta.org.br>>. Acesso em: 16 set. 2012.

PEREIRA, Pedro Augusto de Oliveira. **Tudo sobre o SSH**. Disponível em: < <http://ricardomartins.com.br/2009/04/10/tudo-sobre-o-ssh-parte-1>>. Acesso em: 12 nov. 2012.

PINHEIRO, José Mauricio Santos. **O modelo OSI**. Disponível em: <[http://www.projotoderedes.com.br/artigos/artigo\\_modelo\\_osi.php](http://www.projotoderedes.com.br/artigos/artigo_modelo_osi.php)>. Acesso em: 20 ago. 2012.

SÃO PAULO, Prefeitura. **Telecentros**. Disponível em:< [http://www.prefeitura.sp.gov.br/cidade/secretarias/participacao\\_parce.php](http://www.prefeitura.sp.gov.br/cidade/secretarias/participacao_parce.php)>. Acesso em: 15 set. 2012.

STEWART, Kate. **LTS**. 2012. Disponível em: <<https://wiki.ubuntu.com/LTS>>. Acesso em: 12 out. 2012.

TANEBAUM,A.S. Redes de computadores. Tradução Vandenberg D. de Souza. Rio de Janeiro, Elsevier, 2003, 4ª edição. 920p.

TENDA. **S8**. Disponível em <[www.tendacn.com.br/produtos/linha10100mbps/s8.html](http://www.tendacn.com.br/produtos/linha10100mbps/s8.html)>. Acesso em: 12 out. 2012.

THIN CLIENT. **Intel Enables Thin-But-Not-Pricey Laptops**. Disponível em: < [http://www.thinclient.org/archives/2009/06/intel\\_enables\\_t.html](http://www.thinclient.org/archives/2009/06/intel_enables_t.html) >. Acesso em: 12 out. 2012.

THIN CLIENT. **Intel Enables Thin-But-Not-Pricey Laptops**. Disponível em: < [http://www.thinclient.org/archives/2009/06/intel\\_enables\\_t.html](http://www.thinclient.org/archives/2009/06/intel_enables_t.html) >. Acesso em: 12 out. 2012.

TORRES, Gabriel. **Como o Protocolo TCP/IP Funciona**. Disponível em:<  
<http://www.clubedohardware.com.br/artigos/Como-o-Protocolo-TCP-IP-Funciona-Parte-1/1351/1>>. Acesso em: 20 ago. 2012.

WINDOWS SERVER PASSWORD RECOVERY. **How to boot from Windows Server Password Recovery CD**. Disponível em:  
<<http://www.windowsserverpasswordrecovery.net/boot-from-windows-server-password-recovery-cd.html>>. Acesso em: 03 nov. 2012.

## ANEXOS

### ANEXO A - Teste de memória – Comando free - Servidor LTSP Etapas de 1 a 4 – Thin Client 01

```

Mem:          total        used        free        shared    buffers    cached
-/+ buffers/cache:  257096    1393680
Swap:        1998844         0        1998844

```

```

Mem:          total        used        free        shared    buffers    cached
-/+ buffers/cache:  378812    1271964
Swap:        1998844         0        1998844

```

```

Mem:          total        used        free        shared    buffers    cached
-/+ buffers/cache:  401892    1248884
Swap:        1998844         0        1998844

```

```

Mem:          total        used        free        shared    buffers    cached
-/+ buffers/cache:  434328    1216448
Swap:        1998844         0        1998844

```

ANEXO B - Teste de memória – Comando free - Servidor LTSP  
Etapas de 1 a 4 – Thin Client 02

	total	used	free	shared	buffers	cached
Mem:	1650776	563456	1087320	0	40312	292856
-/+ buffers/cache:		230288	1420488			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650776	866788	783988	0	41636	456808
-/+ buffers/cache:		368344	1282432			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650776	894364	756412	0	41696	468212
-/+ buffers/cache:		384456	1266320			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650776	1109464	541312	0	44888	613208
-/+ buffers/cache:		451368	1199408			
Swap:	1998844	0	1998844			

**ANEXO C - Teste de memória – Comando free - Servidor LTSP  
Etapas de 1 a 4 – Thin Client 01 e 02**

	total	used	free	shared	buffers	cached
Mem:	1650776	493592	1157184	0	39520	238364
-/+ buffers/cache:		215708	1435068			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650776	913280	737496	0	41720	380788
-/+ buffers/cache:		490772	1160004			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650776	1066920	583856	0	43524	478876
-/+ buffers/cache:		544520	1106256			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650776	1176272	474504	0	44700	527712
-/+ buffers/cache:		603860	1046916			
Swap:	1998844	0	1998844			

**ANEXO D - Teste de memória – Comando free - Servidor DRBL**  
**Etapas de 1 a 4 – Thin Client 01**

	total	used	free	shared	buffers	cached
Mem:	1650784	467868	1182916	0	52480	194256
-/+ buffers/cache:		221132	1429652			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650784	720436	930348	0	61652	376412
-/+ buffers/cache:		282372	1368412			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650784	820540	830244	0	62616	474004
-/+ buffers/cache:		283920	1366864			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650784	865252	785532	0	63144	518056
-/+ buffers/cache:		284052	1366732			
Swap:	1998844	0	1998844			

**ANEXO E - Teste de memória – Comando free - Servidor DRBL**  
**Etapas de 1 a 4 – Thin Client 02**

	total	used	free	shared	buffers	cached
Mem:	1650784	467868	1182916	0	52480	194256
-/+ buffers/cache:		221132	1429652			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650784	713120	937664	0	61560	366092
-/+ buffers/cache:		285468	1365316			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650784	826304	824480	0	62444	475508
-/+ buffers/cache:		288352	1362432			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650784	870968	779816	0	63012	519508
-/+ buffers/cache:		288448	1362336			
Swap:	1998844	0	1998844			

ANEXO F - Teste de memória – Comando free - Servidor DRBL  
Etapas de 1 a 4 – Thin Client 01 e 02

	total	used	free	shared	buffers	cached
Mem:	1650784	467868	1182916	0	52480	194256
-/+ buffers/cache:		221132	1429652			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650784	734040	916744	0	66608	386696
-/+ buffers/cache:		280736	1370048			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650784	836876	813908	0	67988	485464
-/+ buffers/cache:		283424	1367360			
Swap:	1998844	0	1998844			

	total	used	free	shared	buffers	cached
Mem:	1650784	884292	766492	0	68788	532300
-/+ buffers/cache:		283204	1367580			
Swap:	1998844	0	1998844			

## ANEXO G - Teste de Processador – Comando vmstat - Servidor LTSP

### Etapas de 2 a 4 – Thin Client 01

```

procs -----memory----- --swap-- -----io----- --system-- -----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1  0    0 1144712 42552 243976  0  0 1703  36 260 610 13  6 65 16
0  0    0 1144712 42552 243976  0  0  0  0  78 176  1  0 99  0
0  0    0 1144588 42552 243980  0  0  0  0 171 337  4  1 95  0
2  0    0 1144588 42552 243980  0  0  0  0 113 214  1  0 99  0
0  0    0 1144588 42560 243980  0  0  0  24  81 188  1  1 98  2
0  0    0 1144588 42560 243980  0  0  0  0 130 262  2  1 98  0
0  0    0 1144588 42560 243980  0  0  0  0  67 148  0  1 99  0
0  0    0 1144588 42560 243980  0  0  0  0  67 156  1  1 99  0
0  0    0 1144588 42560 243980  0  0  0  0  78 175  1  1 99  0
1  0    0 1142936 42620 244048  0  0  72 136 226 687  6  2 78 15
1  0    0 1140068 42636 244396  0  0 388  0 328 641 13  4 79  5
0  1    0 1094140 42772 267016  0  0 22736  0 443 958  9  7 51 34
2  1    0 1087504 42812 269304  0  0 2328  64 1967 1705 51  6 33 11
1  2    0 1104536 43024 273804  0  0 4748  0 4177 4175 60 15  1 24
0  1    0 1096556 43232 276660  0  0 3080  0 3891 4498 26 15 23 35
0  2    0 1085728 43292 281904  0  0 5300  0 4081 4277 19 11 45 25
1  1    0 1075364 43304 294300  0  0 12356  0 3977 4075 14 16 32 38
0  2    0 1065040 43312 304440  0  0 10164  24 3725 3861  2  8 56 35
0  1    0 1060320 43312 308616  0  0 4228  0 4214 3587 31 10 48 11
0  1    0 1052376 43320 312948  0  0 4324  48 4328 3493 37 12 40 11
2  0    0 1049640 43320 315860  0  0 2856  0 4295 4019 46 11 36  7

procs -----memory----- --swap-- -----io----- --system-- -----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1  0    0 1047284 43320 318080  0  0 2196  0 3815 3482 42 10 48  1
0  0    0 1047284 43320 318084  0  0  0  0  63 110  0  1 100  0
0  0    0 1047284 43320 318084  0  0  0  0  43  89  0  0 100  0
0  0    0 1047284 43328 318076  0  0  0  60  61 125  1  0 98  2
0  0    0 1047284 43328 318080  0  0  0  0  50 100  0  1 100  0
0  0    0 1047284 43328 318084  0  0  0  0 150 283  4  1 96  0
0  0    0 1047284 43328 318084  0  0  0  0  44  92  0  0 100  0
0  0    0 1047284 43328 318084  0  0  0  0  55 110  0  0 100  0
0  0    0 1047284 43328 318084  0  0  0  92  70 104  1  1 99  0
0  0    0 1047292 43336 318084  0  0  0  12  68 147  0  0 99  1
0  0    0 1047308 43344 318080  0  0  0  32  79 177  1  0 98  2
0  0    0 1047052 43352 318228  0  0 152  0 162 259  1  1 96  3
0  0    0 1045532 43352 319760  0  0 1516  0 702 775  1  2 92  5
0  0    0 1041168 43352 324064  0  0 4304  4 2181 1505  0  4 82 14
0  0    0 1037944 43352 327252  0  0 3228  0 1548 1160  1  2 86 12
1  1    0 1033268 43352 331884  0  0 4644  0 2034 1376  0  3 81 16
0  0    0 1030716 43360 334176  0  0 2292  32 1159 1022  0  3 85 12
0  0    0 1030584 43360 334344  0  0 128  0 264 396  4  1 95  1
0  0    0 1028848 43360 336080  0  0 1740  4 889 664  0  2 91  7
0  0    0 1026112 43360 338852  0  0 2768  0 1472 1152  1  2 88 10
0  1    0 1021096 43360 342804  0  0 4104  0 2288 1902  4  7 73 17

procs -----memory----- --swap-- -----io----- --system-- -----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2  2    0 1016252 43360 347632  0  0 4740  28 2599 2432  1  4 84 11
2  0    0 1013268 43368 350624  0  0 2928  4 1801 1734  1  3 83 14
0  0    0 1010112 43368 354120  0  0 3496  4 2872 2571  1  5 82 12
1  0    0 1005456 43368 358888  0  0 4768  0 3593 3074  0  4 88  7
1  0    0 1002604 43368 361708  0  0 2872  0 2357 1889  1  4 93  3
0  0    0 999184 43368 365084  0  0 3320  0 3111 2363  0  4 95  1
0  0    0 998192 43376 366176  0  0 1096  32 636 620  0  2 94  5
0  0    0 998192 43376 366180  0  0  0  0  62 122  0  1 100  0
0  0    0 998192 43376 366180  0  0  0  0 133 253  3  1 96  0
0  0    0 998192 43376 366180  0  0  0  0  59 120  0  0 100  0
0  0    0 998192 43376 366180  0  0  0  0  48  94  1  0 100  0
2  0    0 998192 43376 366180  0  0  0  0  71 139  1  1 99  0
0  0    0 998192 43384 366180  0  0  0  12  72 129  0  0 99  1
0  0    0 998192 43384 366180  0  0  0  0  73 133  0  1 99  0
0  0    0 996068 43384 367584  0  0 1400  0 958 996 10  3 83  4
0  0    0 996068 43384 367584  0  0  0  0 105 174  1  0 99  0
0  0    0 996068 43384 367584  0  0  0  0  66 125  0  1 99  0
0  0    0 996068 43384 367584  0  0  0  0  73 130  0  1 99  0
0  0    0 996068 43392 367576  0  0  0  20  56 102  0  1 98  1
1  0    0 995120 43392 367644  0  0  52  0 478 729 10  6 82  2
4  0    0 994764 43404 367932  0  0 408  0 1472 2377 20 17 57  6

procs -----memory----- --swap-- -----io----- --system-- -----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1  0    0 986624 43516 374548  0  0 6640  4 2011 3631 41 19 23 17
6  1    0 975744 43548 375104  0  0  564  16 1988 3746 55 12 30  4

```

```

4 2      0 954968 43660 376036      0      0 988 228 3368 7730 76 20 3 2
3 0      0 934584 43688 376560      0      0 572 0 2510 4508 84 16 1 0
10 0     0 902924 43788 376596      0      0 36 624 1974 3138 90 10 0 0
2 0      0 894876 43800 376612      0      0 12 0 1931 2944 93 7 0 0
0 1      0 886960 43800 376856      0      0 256 4 5500 2406 75 8 15 2
1 0      0 898692 43832 377640      0      0 808 144 4808 3405 54 11 29 5
5 0      0 876916 43868 379500      0      0 1784 64 1058 3230 91 8 1 0
3 0      0 879704 43876 379596      0      0 104 0 2072 2493 81 8 11 0
0 0      0 879724 43876 379596      0      0 0 0 664 1178 11 3 87 0
0 0      0 878536 43876 379596      0      0 0 0 506 1085 13 3 85 0
1 1      0 873092 43892 379580      0      0 0 144 421 974 20 3 72 5
0 0      0 870860 43936 379536      0      0 4 72 634 982 15 3 71 10
0 0      0 870876 43936 379596      0      0 0 0 68 138 1 0 100 0
0 0      0 870876 43936 379596      0      0 0 0 45 97 0 0 100 0
0 0      0 870876 43936 379596      0      0 0 0 57 118 0 1 99 0
0 0      0 870752 43936 379596      0      0 0 0 137 272 3 1 97 0
0 0      0 864720 43972 379560      0      0 28 140 928 2075 30 7 62 1
0 0      0 864720 43972 379560      0      0 0 0 61 141 1 0 100 0
0 0      0 864736 43972 379604      0      0 0 0 55 137 0 0 100 0
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free  buff  cache      si  so  bi  bo  in  cs  us  sy  id  wa
1 0      0 864768 43972 379604      0      0 0 0 71 147 0 0 100 0
0 0      0 864768 43972 379604      0      0 0 0 60 140 0 0 99 0
0 0      0 863124 43980 379604      0      0 0 84 678 1233 16 6 76 2
0 0      0 863132 43980 379604      0      0 0 0 53 127 0 0 100 0
0 0      0 863140 43980 379604      0      0 0 0 64 159 1 0 100 0
0 0      0 863172 43980 379604      0      0 0 4 59 135 0 1 100 0
0 0      0 863180 43980 379604      0      0 0 0 63 137 0 1 100 0
0 0      0 863180 43980 379604      0      0 0 0 49 105 1 0 100 0
0 0      0 863164 43988 379604      0      0 0 32 164 309 3 2 94 2
0 0      0 863164 43988 379604      0      0 0 0 107 200 1 1 99 0
0 0      0 863164 43992 379600      0      0 4 28 71 136 0 0 99 1
0 1      0 862504 44016 379596      0      0 68 0 147 247 2 1 95 2
0 1      0 835360 44048 406140      0      0 26576 0 1378 1650 7 5 49 39
0 1      0 803320 44056 438108      0      0 31888 0 167 469 1 3 51 45
0 1      0 789872 44136 448772      0      0 10828 64 494 1018 13 7 38 42
1 0      0 769572 44296 462148      0      0 13552 300 610 900 20 4 49 27
2 0      0 765532 44516 463344      0      0 1360 0 706 1230 16 6 50 27
0 1      0 757068 44524 468172      0      0 4832 24 558 854 23 3 53 21
0 1      0 754036 44640 469072      0      0 1052 0 407 843 24 2 50 24
1 0      0 743200 44648 476860      0      0 7748 140 1993 2779 38 7 36 20
1 0      0 742168 44656 477436      0      0 560 136 1694 1784 51 8 37 5
r b      swpd  free  buff  cache      si  so  bi  bo  in  cs  us  sy  id  wa
1 0      0 741928 44656 477496      0      0 20 24 374 286 4 1 95 1
0 0      0 741812 44656 477496      0      0 0 0 103 183 1 1 99 0
0 0      0 741812 44656 477496      0      0 0 0 76 135 1 1 99 0
0 0      0 741812 44664 477488      0      0 0 80 99 185 1 0 97 2
0 0      0 741812 44664 477496      0      0 0 4 87 150 0 0 100 0
0 0      0 741812 44664 477496      0      0 0 0 98 172 1 1 98 0
0 0      0 741812 44664 477496      0      0 0 0 73 136 1 0 99 0
0 0      0 741812 44664 477496      0      0 0 0 88 156 0 0 100 0
0 0      0 741812 44664 477496      0      0 0 0 83 146 1 1 99 0
0 0      0 741812 44672 477496      0      0 0 28 86 160 0 0 98 1
0 0      0 741812 44672 477496      0      0 0 0 82 147 0 0 100 0
1 0      0 741820 44672 477496      0      0 0 0 107 209 1 1 99 0
0 0      0 741820 44672 477496      0      0 0 0 95 172 1 1 99 0
0 0      0 741828 44672 477496      0      0 0 0 104 169 0 0 100 0
1 0      0 741828 44672 477496      0      0 0 0 82 135 1 0 99 0
0 0      0 741828 44680 477496      0      0 0 16 92 168 0 1 98 2
0 0      0 741828 44680 477496      0      0 0 0 82 146 0 0 100 0
1 0      0 741828 44680 477496      0      0 0 0 92 162 1 0 99 0
0 0      0 741828 44680 477496      0      0 0 0 128 217 2 0 99 0
0 0      0 741828 44680 477496      0      0 0 0 95 171 0 0 100 0
1 1      0 709096 44680 509796      0      0 32320 0 176 349 1 2 87 11
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free  buff  cache      si  so  bi  bo  in  cs  us  sy  id  wa
1 1      0 702800 44708 510440      0      0 668 16 1409 1918 22 8 30 41
2 0      0 688148 44848 513820      0      0 3576 0 671 1580 28 6 42 24
2 0      0 673096 45024 515768      0      0 2116 0 795 1444 74 7 8 12
4 0      0 663284 45060 518640      0      0 2556 120 1636 2206 74 7 3 17
4 0      0 657648 45064 518960      0      0 280 0 1394 2893 81 8 10 1
1 0      0 650112 45076 524448      0      0 5452 4 2327 1025 20 6 54 20
1 0      0 649896 45076 526136      0      0 1696 0 392 426 40 11 44 5
3 0      0 648996 45084 526232      0      0 128 140 229 297 14 7 76 3
1 0      0 648996 45084 526264      0      0 0 224 150 213 2 1 98 0
0 0      0 652096 45084 526264      0      0 0 0 145 244 4 1 95 0
1 0      0 652096 45084 526264      0      0 0 0 149 281 1 1 98 0

```

ANEXO H - Teste de Processador – Comando vmstat - Servidor LTSP  
Etapas de 2 a 4 – Thin Client 02

```

procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1  0    0 1145136 42152 243960  0  0 1716  34 258 617 13  6 66 15
0  0    0 1145136 42152 243972  0  0  0  0  85 170  1  1 99  0
1  0    0 1145128 42152 243988  0  0  0  0 118 232  1  0 99  0
0  0    0 1145128 42152 243988  0  0  0  0  87 185  1  0 99  0
0  0    0 1145128 42152 243988  0  0  0  0  56 125  0  1 100  0
0  0    0 1143128 42172 244068  0  0 100  0 356 892 15  4 80  3
0  1    0 1139112 42312 244436  0  0 600 32 204 370 11  2 80  7
1  0    0 1091000 42320 268356  0  0 23864 0 521 1048 14  7 50 30
1  1    0 1085760 42336 269576  0  0 1280 0 468 570 53  4 34 10
1  2    0 1104888 42600 270728  0  0 1416 0 660 893 51  7 10 33
0  1    0 1099756 42788 272552  0  0 2056 0 679 1656 26  6 37 31
1  2    0 1088292 42840 279168  0  0 6628 72 422 970 12  5 36 47
1  1    0 1080640 42856 289052  0  0 9836 0 2169 2318 14 10 36 40
1  0    0 1072564 42856 297044  0  0 8000 0 5051 5053 15  9 51 26
2  0    0 1062892 42856 303172  0  0 6088 0 5180 4638 29 12 43 16
2  0    0 1058908 42856 306840  0  0 3644 0 5741 4450 48 16 33  2
1  0    0 1055552 42856 310140  0  0 3384 0 5673 4814 53 10 36  1
0  0    0 1051940 42864 313632  0  0 3436 56 5134 5265 13  6 79  1
0  0    0 1048584 42864 316960  0  0 3328 0 4960 5038  2  6 92  0
0  0    0 1047468 42864 318060  0  0 1104 0 1984 2058  1  1 98  0
0  0    0 1047476 42864 318080  0  0  0  0  61 114  0  1 99  0
procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1  0    0 1047476 42864 318080  0  0  0  0  48  96  0  0 99  0
0  0    0 1047476 42864 318080  0  0  0  0  56 105  0  0 100  0
0  0    0 1047476 42872 318080  0  0  0 20  59 111  0  1 97  2
0  0    0 1047476 42872 318080  0  0  0  0  61 119  0  1 100  0
0  0    0 1047476 42872 318080  0  0  0  0  48 104  1  0 100  0
0  0    0 1047484 42872 318080  0  0  0  0  67 147  0  1 100  0
0  0    0 1045180 42888 320168  0  0 2148 28 1159 1297  2  2 86 10
0  1    0 1039240 42888 326068  0  0 5848 0 2447 2215  1  4 74 22
1  0    0 1032760 42888 332352  0  0 6336 80 2997 2314  0  4 74 22
0  1    0 1026792 42888 338340  0  0 5972 0 3347 2381  1  3 75 22
0  0    0 1021964 42888 342720  0  0 4420 0 2775 2556  7  7 68 18
1  0    0 1017852 42888 346968  0  0 4252 0 2492 2332  1  5 84 11
0  0    0 1014496 42896 350400  0  0 3356 24 1963 1921  0  4 84 12
0  0    0 1009832 42896 355020  0  0 4636  4 3240 3066  0  6 78 16
0  0    0 1003180 42896 361500  0  0 6504 0 4989 4425  1  6 87  6
0  0    0 998344 42896 366480  0  0 4916 0 3818 3419  0  4 92  4
0  0    0 998220 42896 366464  0  0  0  0 132 252  3  2 96  0
0  0    0 998220 42896 366464  0  0  0  0  57 115  0  0 100  0
0  0    0 998096 42904 366464  0  0  0 36  64 121  0  0 98  2
0  0    0 998096 42904 366464  0  0  0  0  67 118  1  0 100  0
0  0    0 998096 42904 366464  0  0  0  0  45  91  0  1 100  0
procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2  0    0 997228 42904 366964  0  0 740  0 338 374  4  2 92  2
0  0    0 995880 42904 367884  0  0 644  0 612 662  3  1 94  2
0  0    0 995840 42904 367852  0  0  0  0  72 128  0  0 99  0
0  0    0 995840 42912 367844  0  0  0 36  60 116  0  0 97  3
0  0    0 995840 42912 367852  0  0  0  0  69 132  0  1 99  0
0  0    0 995840 42912 367856  0  0  0  0  45  93  0  0 100  0
1  1    0 994924 42924 367948  0  0 108  0 1110 1651 16 18 62  4
2  0    0 986992 42948 374644  0  0 6816 0 1913 3091 35 16 35 15
3  0    0 980840 43060 374668  0  0 172  8 1856 3623 45 18 33  5
3  0    0 968940 43096 374812  0  0  36 192 2356 6592 75 14 10  2
5  0    0 940496 43120 375228  0  0 404 72 3314 5141 81 18  0  0
2  0    0 925952 43216 375544  0  0 292 576 3630 2392 80 10  5  5
7  1    0 907088 43228 375512  0  0 16  0 3479 2791 86 12  2  0
4  1    0 900956 43244 375528  0  0  8  92 7738 3709 51 11 38  1
1  0    0 901960 43256 375520  0  0  8  60 3863 2660 58  9 32  1
2  0    0 886816 43268 376124  0  0 560  0 1438 4500 80 10 11  0
0  0    0 886204 43268 376116  0  0  0  0  472 1700 25  2 73  0
0  0    0 886300 43268 376116  0  0  0  0  174 316  3  2 96  0
0  0    0 879128 43268 376116  0  0  0  0  645 1544 29  4 67  0
0  0    0 879168 43268 376116  0  0  0  0  173 296  2  2 96  0
0  0    0 877548 43280 376112  0  0  4  72 534 852 13  4 82  2
procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1  0    0 877532 43280 376116  0  0  0  0  97 181  1  1 99  0

```

```

0 0      0 877516 43280 376116    0    0    0 224 125 146 1 1 99 0
0 1      0 872432 43288 376708    0    0 632    0 242 488 9 3 81 8
0 0      0 875452 43300 377160    0    0 432    0 335 787 8 3 85 4
0 0      0 869348 43340 377124    0    0 28 104 847 1957 27 6 65 2
0 0      0 869284 43340 377172    0    0    0    0 86 184 0 0 99 0
0 0      0 869316 43340 377172    0    0    0    4 94 177 0 0 100 0
0 0      0 869332 43340 377172    0    0    0    0 97 202 1 1 99 0
0 0      0 869380 43340 377172    0    0    0    0 90 174 0 0 100 0
0 0      0 867404 43340 377172    0    0    0    0 599 1111 11 6 83 0
0 0      0 867380 43348 377172    0    0    0 72 183 332 2 1 95 2
0 0      0 867420 43348 377172    0    0    0    4 88 156 1 0 99 0
0 0      0 867420 43348 377172    0    0    0    0 94 197 0 0 100 0
0 0      0 867452 43348 377172    0    0    0    0 82 145 0 0 100 0
0 0      0 867420 43348 377172    0    0    0    0 84 153 1 0 100 0
0 0      0 867420 43348 377172    0    0    0    0 76 141 1 0 99 0
1 0      0 867420 43356 377172    0    0    0 36 102 184 0 0 98 2
0 0      0 867420 43356 377172    0    0    0    0 72 135 0 0 100 0
0 0      0 867420 43356 377172    0    0    0    0 89 166 0 0 99 0
1 1      0 849404 43412 394316    0    0 17244 0 1392 1545 9 6 54 31
0 1      0 812876 43420 430776    0    0 36480 0 197 515 1 4 51 45

procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd   free   buff  cache    si   so    bi   bo    in   cs us sy id wa
1 1      0 798436 43452 443800    0    0 13120 68 370 761 10 5 49 37
1 1      0 776440 43648 459380    0    0 15656 0 443 809 16 4 50 30
0 1      0 770524 43864 460764    0    0 1696 0 858 1452 20 6 49 26
0 1      0 764284 43940 464136    0    0 3324 280 629 1008 19 3 37 41
1 0      0 759152 44052 466744    0    0 2684 0 418 844 23 3 51 24
0 0      0 756292 44072 467656    0    0 916 32 2408 2816 31 8 52 10
0 0      0 756276 44072 467668    0    0    0 16 271 275 2 1 97 0
0 0      0 756284 44072 467668    0    0    0 0 115 193 0 0 100 0
0 0      0 756284 44072 467668    0    0    0 0 166 270 1 1 98 0
0 0      0 756284 44080 467668    0    0    0 120 127 213 1 1 97 3
0 0      0 756284 44080 467668    0    0    0 0 104 172 0 0 100 0
0 0      0 756284 44080 467672    0    0    0 0 123 212 1 1 99 0
0 0      0 756284 44080 467672    0    0    0 0 106 181 1 0 99 0
0 0      0 756284 44080 467672    0    0    0 0 114 189 1 1 99 0
0 0      0 756284 44080 467672    0    0    0 0 103 173 0 1 99 0
0 0      0 756284 44088 467672    0    0    0 204 157 203 0 1 98 2
0 0      0 756284 44088 467672    0    0    0 0 108 177 1 0 99 0
0 0      0 756284 44088 467672    0    0    0 0 129 219 1 0 99 0
0 0      0 756300 44088 467672    0    0    0 0 121 197 1 1 99 0
0 0      0 756300 44088 467672    0    0    0 0 106 173 1 1 99 0
0 0      0 756300 44088 467672    0    0    0 4 108 185 1 1 99 0

procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd   free   buff  cache    si   so    bi   bo    in   cs us sy id wa
0 0      0 756300 44096 467672    0    0    0 24 122 208 1 0 98 2
0 0      0 756308 44096 467672    0    0    0 0 118 198 0 1 99 0
0 0      0 756308 44096 467672    0    0    0 0 99 161 1 1 99 0
0 0      0 756308 44096 467672    0    0    0 0 115 187 1 0 100 0
0 0      0 756308 44096 467672    0    0    0 0 105 176 0 1 99 0
1 0      0 756308 44096 467672    0    0    0 0 114 194 1 0 99 0
0 0      0 756308 44104 467672    0    0    0 16 114 195 0 1 97 2
1 0      0 721012 44104 499976    0    0 32320 0 684 944 14 6 47 33
1 1      0 707848 44256 502588    0    0 2780 0 1243 1936 26 8 43 23
1 0      0 692700 44292 505588    0    0 3008 0 781 1274 36 7 38 20
2 0      0 679980 44472 508140    0    0 2444 96 941 1318 68 7 7 19
1 0      0 675956 44484 508528    0    0 420 0 4058 3690 81 10 7 3
0 0      0 674848 44484 508880    0    0 144 0 471 940 56 1 43 0
0 0      0 673360 44492 508928    0    0 128 80 180 314 2 2 93 4
1 0      0 673360 44492 508964    0    0    0 16 116 206 1 0 99 0
0 0      0 673360 44492 508964    0    0    0 0 121 211 0 0 100 0
0 0      0 676088 44492 508964    0    0    0 0 184 333 6 1 93 0
0 0      0 676088 44492 508964    0    0    0 0 175 301 1 0 99 0

```

ANEXO I - Teste de Processador – Comando vmstat - Servidor LTSP  
Etapas de 2 a 4 – Thin Client 01 e 02

```

procs -----memory----- --swap-- -----io----- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1  0    0 1137368 42160 252424  0  0 2016  39 366 759 15  7 58 20
0  0    0 1134128 42160 255780  0  0 3328  0 5024 5051  2  5 93  0
0  0    0 1130780 42160 259108  0  0 3328  0 4885 4917  1  4 95  0
3  0    0 1127548 42168 262272  0  0 3200  52 4863 4889  0  6 92  2
0  0    0 1125316 42168 264508  0  0 2256  0 3714 3772  1  5 95  0
0  0    0 1125316 42168 264540  0  0  0  0  93 152  1  0 100  0
1  0    0 1124936 42176 264540  0  0  0  28 3148 3156  1  9 88  3
0  0    0 1124936 42176 264540  0  0  0  0 3502 3560  1  5 94  0
1  0    0 1124952 42176 264540  0  0  0  0 3500 3589  2  4 94  0
0  0    0 1124952 42176 264540  0  0  0  0 3527 3621  1  6 93  0
0  0    0 1124952 42176 264540  0  0  0  4 3564 3656  1  8 91  0
0  0    0 1124952 42184 264540  0  0  0  32 3561 3666  1  5 92  2
1  0    0 1124952 42184 264540  0  0  0  0 3528 3595  1  6 93  0
0  0    0 1124952 42184 264540  0  0  0  0 3292 3369  2  6 92  0
0  0    0 1124952 42184 264540  0  0  0  0 2968 3058  1  6 93  0
0  0    0 1124960 42184 264540  0  0  0  0  91 164  1  0 99  0
0  0    0 1120840 42200 268604  0  0 4092  28 1930 1817  1  3 75 22
1  0    0 1115428 42200 273876  0  0 5272  0 2373 2403  1  5 77 18
1  0    0 1106564 42220 280544  0  0 6792  0 3635 2669 15  6 53 27
0  1    0 1098276 42352 285856  0  0 5356  0 2994 2103 12  6 51 31
1  1    0 1074040 42356 300612  0  0 14744  0 2065 1989  6  9 42 43

procs -----memory----- --swap-- -----io----- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1  2    0 1049876 42388 312412  0  0 11872  24 2789 2605  9  9 31 51
1  3    0 1034544 42612 320504  0  0 8292  0 1968 2174  8  6 42 44
0  3    0 1028560 42760 324588  0  0 4216  0 1704 1934 12  5 37 46
2  1    0 1016396 42796 331408  0  0 6804  0 3654 3343 32  9  6 53
0  3    0 1003468 42804 344156  0  0 12788  28 3337 2998  3  4 48 45
2  1    0 992104 42804 355092  0  0 10992  0 5260 3447 41  7  8 44
1  1    0 1007844 42840 361728  0  0 6736  0 4197 3574 58  8  3 32
1  0    0 1000504 42860 365228  0  0 3356  48 3429 2562 32  5 38 24
1  0    0 999716 42860 366280  0  0 1088  0 3203 2925 51  7 38  4
2  0    0 993648 42884 367616  0  0 1360  64 2940 2379 65  9  9 17
0  0    0 996268 42908 368012  0  0 460  0 1710 3023 22  9 66  3
0  0    0 996244 42908 368080  0  0  0  0  92 134  1  0 100  0
0  0    0 996048 42916 368072  0  0  0  80 2178 1684  4  4 90  2
2  0    0 995768 42916 368080  0  0  0  0 2091 1953 11  7 82  0
1  0    0 984956 42952 369564  0  0 1540  4 5728 4753 35 15 38 12
1  0    0 991864 43128 370452  0  0 1072  0 4380 3612 20  7 55 17
0  0    0 991968 43128 370480  0  0  0  0 3269 2643  1  4 95  0
0  0    0 990456 43128 371780  0  0 1384  0 4731 3855  9  6 82  3
1  0    0 990424 43136 371868  0  0  0  40 2458 2099  0  3 95  3
0  0    0 990464 43136 371868  0  0  0  0 1771 1558  1  2 98  0
0  0    0 990464 43136 371868  0  0  0  0  74 144  0  0 99  0

procs -----memory----- --swap-- -----io----- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2  0    0 990464 43136 371868  0  0  0  0  89 182  0  1 99  0
0  1    0 990216 43136 371868  0  0  16  0  74 146  1  1 98  1
0  0    0 989028 43148 372192  0  0  380  0 1522 2102 16 24 54  6
3  0    0 981220 43196 378660  0  0 6548  76 2550 4459 45 18 21 15
4  1    0 970620 43244 378784  0  0  68  8 2221 3899 52 14 30  5
8  0    0 949044 43288 378996  0  0 296  204 3270 7348 78 21  2  0
3  0    0 928648 43296 379504  0  0 404  0 2431 3669 90 10  0  0
8  0    0 908856 43388 379448  0  0 24  540 3854 2936 75 15  7  4
3  0    0 902768 43408 379500  0  0 24  0 3529 3852 85 12  3  0
4  0    0 892540 43420 379500  0  0 12  140 6292 3783 49 25 20  7
3  0    0 883652 43468 379476  0  0 56  84 4116 4906 71 18 10  1
6  1    0 875544 43500 379924  0  0 420  4 4011 5986 72 21  7  1
11 0    0 861212 43636 380920  0  0 1096  0 1710 5922 84 16  0  0
4  2    0 846748 43676 381396  0  0 524  216 2707 7562 80 19  0  0
6  0    0 828496 43700 381772  0  0 432  0 2711 4825 84 14  2  0
7  0    0 800416 43788 381872  0  0 20  596 2008 3075 89 11  0  0
6  0    0 779008 43808 381852  0  0 24  0 1303 4372 87 14  0  0
2  0    0 778504 43808 381880  0  0  0  0 2321 1542 95  4  0  0
2  0    0 771064 43808 381880  0  0  0  0 7423 4044 61 14 25  0
8  0    0 765712 43836 382544  0  0 716  100 1967 2494 65  7 26  3
11 0    0 751648 43876 383800  0  0 1196  88 2175 3649 88 12  0  0

procs -----memory----- --swap-- -----io----- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
6  0    0 756804 43880 383840  0  0  4  0 1525 3516 58  9 34  0
0  0    0 756256 43880 383840  0  0  0  0  483 1086 10  3 87  0

```

```

0 0      0 748116 43884 383836    0    0    4   280  703 1396 29  6 65  0
0 0      0 748132 43884 383840    0    0    4     0  286  520  4  2 93  1
0 0      0 746148 43884 383844    0    0    0     0  575 1025 11  5 84  0
0 0      0 746156 43892 383844    0    0    0     76  143  264  1  0 98  1
0 0      0 746040 43892 383844    0    0    0     0  125  258  2  1 98  0
0 0      0 740380 43920 383816    0    0   28    96  778 1592 24  6 69  1
0 0      0 739892 43920 383852    0    0    0     0  567  890 10  5 86  0
1 0      0 739940 43920 383852    0    0    0     0  195  391  4  1 96  0
0 0      0 739972 43920 383852    0    0    0     0  108  219  0  0 99  0
0 0      0 740020 43920 383852    0    0    0     0  167  309  2  1 97  0
1 0      0 738484 43920 383852    0    0    0    20  616 1093 11  6 83  0
0 0      0 738484 43928 383852    0    0    0   100  109  205  0  1 97  2
1 0      0 737980 43968 383912    0    0   280     0  207  349  1  2 93  4
0 2      0 712604 43992 404824    0    0 20788     0 1393 1566 15  6 44 36
0 2      0 677200 44008 438420    0    0 33652     0  309  743  6  4 43 47
0 1      0 665968 44032 451804    0    0 13432     4  577 1166 15  8 46 32
1 1      0 646344 44272 466900    0    0 15200   156  466  871 13  4 39 45
1 0      0 641056 44364 468232    0    0  1404     40  706 1267 18  5 43 34
0 1      0 632460 44528 472624    0    0  4644     16  753 1113 21  4 47 28
procs  -----memory-----  ---swap---  -----io-----  -system-  -----cpu-----
r  b      swpd   free   buff  cache    si  so   bi   bo   in  cs  us  sy  id  wa
1  1      0 628576 44644 474264    0  0  1700     0  482  967 28  3 48 22
3  0      0 624436 44668 475036    0  0   776    48 2324 2882 33  9 49  9
1  0      0 614140 44740 475132    0  0   100    16 2028 2276 40 12 46  3
0  1      0 607760 44792 475120    0  0    92   256  961 1186 45  6 47  3
1  0      0 596312 44808 482748    0  0  7536     0 1909 2835 35  7 37 21
1  0      0 595676 44812 482828    0  0   132    24 1764 1881 57  7 36  1
0  0      0 595428 44812 482880    0  0    20    24  505  527  7  1 92  1
0  0      0 595436 44812 482880    0  0     0     0  155  253  0  1 99  0
0  0      0 595436 44812 482880    0  0     0     0  145  216  1  0 99  0
0  0      0 595436 44820 482880    0  0     0     0  112  169  277  1  1 96  3
0  0      0 595436 44820 483080    0  0   184     0  236  312  1  1 98  1
1  0      0 595436 44820 483068    0  0     0   108  172  257  0  1 99  0
0  0      0 595444 44820 483068    0  0     0     0  237  373  4  1 96  0
0  0      0 595320 44820 483068    0  0     0     0  154  261  1  1 99  0
1  0      0 595328 44820 483068    0  0     0     0  140  223  1  1 99  0
0  0      0 595328 44828 483068    0  0     0    28  159  267  1  0 97  2
0  0      0 595328 44828 483068    0  0     0   272  190  234  1  1 99  0
1  0      0 595336 44828 483068    0  0     0     0  144  230  0  0 100  0
0  0      0 595336 44828 483068    0  0     0     0  134  228  0  0 100  0
1  0      0 595344 44828 483068    0  0     0     0  151  257  1  1 99  0
1  0      0 595228 44828 483068    0  0     0     0  193  321  2  1 97  0
procs  -----memory-----  ---swap---  -----io-----  -system-  -----cpu-----
r  b      swpd   free   buff  cache    si  so   bi   bo   in  cs  us  sy  id  wa
1  0      0 595228 44836 483068    0  0     0    16  158  263  0  1 98  1
0  0      0 595228 44836 483068    0  0     0     0  142  220  1  0 99  0
0  1      0 589608 44836 485968    0  0  2912     0  709 1110 15  5 72  8
0  1      0 555672 44860 516064    0  0 30076     0 1101 1576 15  7 44 34
1  0      0 541748 44996 519480    0  0 3560     0  630 1248 22  5 47 27
1  1      0 524488 45172 521372    0  0 2092     0  863 1349 65  6 19 10
7  1      0 517728 45212 523760    0  0 2052   124 1531 2385 73  8  1 18
2  0      0 514560 45224 523996    0  0   276     0 3088 2271 73  8 17  2
2  0      0 513436 45224 524276    0  0   144     0  445  847 37  8 55  1
0  0      0 512072 45232 524324    0  0   128    92  189  336  2  1 94  4
1  0      0 506520 45244 524348    0  0    20    16 1079 1524 23  4 72  2
1  0      0 485348 45264 525064    0  0   776     0 1266 1915 46  7 42  5
3  0      0 476344 45280 526508    0  0  1184   128 2206 3049 55 10 26  9
1  0      0 470616 45288 526832    0  0   276     0 2693 1638 47  6 42  5
1  1      0 461596 45288 533068    0  0  6276     4  583  927 56  4 23 18
1  0      0 460852 45296 534000    0  0   864   176  697 1047 85 12  2  2
0  0      0 463580 45296 534008    0  0    64     0  329  776 19  2 78  1
2  0      0 467540 45296 534072    0  0     0     0  165  276  4  0 96  0
0  0      0 467540 45296 534072    0  0     0     0  159  273  2  1 97  0
0  0      0 467540 45296 534072    0  0     0     0  167  289  1  1 98  0
1  0      0 467556 45296 534072    0  0     0     0  163  284  1  1 98  0
procs  -----memory-----  ---swap---  -----io-----  -system-  -----cpu-----
r  b      swpd   free   buff  cache    si  so   bi   bo   in  cs  us  sy  id  wa
1  0      0 470300 45304 534072    0  0     0    88  273  498  5  1 93  2
0  0      0 470308 45304 534072    0  0    64     0  175  343  2  1 97  1
2  0      0 470316 45304 534136    0  0     0     0  142  245  1  1 99  0
0  0      0 470316 45304 534136    0  0     0     0  177  318  2  1 98  0

```

ANEXO J - Teste de Processador – Comando vmstat - Servidor DRBL  
Etapas de 2 a 4 – Thin Client 01

```

procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
0  0    0 1033960 55076 286472  0  0  814  14  372  976 11  5  77  7
0  0    0 1033952 55076 286500  0  0  0  0  769 1648  4  3  93  0
0  0    0 1033936 55076 286500  0  0  0  0  631 1336  4  1  95  0
0  0    0 1033936 55076 286500  0  0  0  0  180 1212  0  0 100  0
0  0    0 1033936 55076 286500  0  0  0  0  190 1177  1  0  99  0
0  0    0 1033904 55076 286500  0  0  0  0  296 1399  5  1  94  0
0  0    0 1033904 55084 286500  0  0  0  32  202 1251  0  1  94  5
0  0    0 1033904 55084 286500  0  0  0  0  177 1238  1  1  99  0
0  0    0 1033780 55084 286500  0  0  0  0  236 1297  1  1  98  0
0  0    0 1033780 55084 286500  0  0  0  0  231 1259  0  1  99  0
0  0    0 1033780 55092 286492  0  0  0  40  712 1900  5  4  86  5
0  0    0 1033656 55092 286500  0  0  0  0  257 1376  5  1  95  0
5  0    0 1033648 55092 286500  0  0  0  0  4156 6270 13 18  69  0
2  0    0 1033648 55092 286500  0  0  0  0  4768 7473 18 18  64  0
4  0    0 1033648 55092 286500  0  0  0  0  4784 7908 18 22  60  0
1  0    0 1033648 55100 286492  0  0  0  48  5021 7937 19 20  57  4
1  0    0 1033648 55100 286500  0  0  0  0  5320 8198 16 17  67  0
2  0    0 1033648 55100 286500  0  0  0  0  5264 8385  5 17  78  0
2  0    0 1033648 55100 286500  0  0  0  0  5400 8428  7 18  75  0
1  0    0 1033624 55100 286500  0  0  0  0  5207 7265 16 19  65  0
0  0    0 1033632 55100 286500  0  0  0  8  4771 7254 16 23  60  0
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2  0    0 1033632 55108 286500  0  0  0  16  4836 8014 15 19  64  2
1  0    0 1033632 55108 286500  0  0  0  0  4789 7969 14 21  65  0
0  0    0 1033632 55108 286500  0  0  0  0  3150 5825 11 11  78  0
1  0    0 1033640 55108 286500  0  0  0  0  293 1409  3  1  96  0
2  0    0 1033640 55108 286500  0  0  0  0  208 1206  1  0  99  0
0  0    0 1033640 55108 286500  0  0  0  4  190 1252  1  1  99  0
0  0    0 1033640 55116 286496  0  0  0  16  180 1220  0  1  94  4
0  0    0 1033640 55116 286500  0  0  0  0  186 1125  0  0 100  0
0  0    0 1033640 55116 286500  0  0  0  0  199 1216  1  0  99  0
0  0    0 1033640 55116 286500  0  0  0  0  233 1310  1  1  98  0
0  0    0 1033516 55124 286492  0  0  0  32  424 4949  2  1  92  5
1  0    0 1032804 55124 287200  0  0  764  0  4686 7644 22 14  58  5
0  0    0 1032560 55132 287376  0  0  120  0  930 1324  7  6  85  3
0  1    0 1031900 55260 287748  0  0  628  0  1317 2363  9  7  76  8
0  0    0 1030872 55304 288832  0  0 1012 100 2843 3732 12 17  58 13
2  0    0 1030152 55372 289468  0  0  624 120 1758 3304  9  9  70 12
0  0    0 1028972 55508 290384  0  0 1136  0  2892 4174 11 18  59 12
2  0    0 1025328 55724 292888  0  0 2712 152 6371 6872 15 31  23 31
1  1    0 1025336 55936 293552  0  0  512 620 2913 4064 14 18  40 28
0  0    0 1024696 56036 293908  0  0  488 260 5735 6383 14 30  43 13
0  0    0 1021280 56180 297108  0  0 3332 112 6724 7042 19 31  27 23
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
3  4    0 1016652 56340 301100  0  0  4048 272 7151 9182 15 28  34 23
1  1    0 1015448 56488 302416  0  0 1424 360 6570 10005 13 34  27 27
2  3    0 1013288 56596 304760  0  0 2228 212 6831 6325 13 32  34 20
0  2    0 1011904 56680 305648  0  0  556 524 4337 5962 12 22  34 33
1  0    0 1010884 56748 306412  0  0  796 384 3189 4222 13 18  50 19
4  0    0 1010244 56816 307692  0  0 1316  0  7180 7804 19 35  37  9
1  1    0 1007948 56880 309840  0  0 2180 100 7077 7015 14 35  39 13
3  0    0 1006976 56968 310304  0  0  528 144 3604 5723 14 19  50 16
1  1    0 1003736 57060 313748  0  0 3504 140 11125 18831 15 39  31 15
0  1    0 1002880 57220 314184  0  0  516 492 8017 10413 17 34  23 26
3  0    0 1001076 57252 315864  0  0 1500 156 7586 10995 14 31  45 10
2  0    0 999940 57284 316964  0  0 1080  36 5587 6632 16 27  48  9
2  1    0 998504 57344 317556  0  0  652  32 5396 5537 22 29  41  7
0  0    0 998100 57444 317664  0  0  236 184 4341 4827 17 21  42 20
3  0    0 996840 57516 318188  0  0  476 188 11547 19401 19 33  41  6
3  1    0 996532 57616 319040  0  0  776 160 7233 7391 15 35  38 12
3  1    0 996068 57820 319416  0  0  436 564 4772 5839 16 25  33 27
2  0    0 993220 58024 321172  0  0 1644 636 8317 12277 17 26  27 30
3  1    0 988012 58064 327964  0  0 6904 152 18308 56462 15 44  31 10
0  2    0 986592 58384 328220  0  0  348 628 5364 6669  8 23  33 36
1  1    0 986056 58608 328472  0  0  368 708 5575 6256  8 26  24 41
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2  2    0 985764 58852 328624  0  0  164 772 6336 6811  8 27  33 32

```

```

0 2      0 985496 58988 328704      0 0      52 476 3257 5055 3 13 58 26
1 0      0 984796 59024 329204      0 0      516 72 8821 7255 10 31 49 10
1 0      0 982644 59164 331816      0 0      2620 212 8377 7885 15 37 27 21
0 1      0 982060 59216 331904      0 0      48 84 3913 5067 14 21 57 8
4 0      0 981184 59376 332140      0 0      280 240 4764 5259 15 24 41 20
0 0      0 980912 59520 332160      0 0      144 28 1670 3366 9 10 69 11
0 0      0 980704 59644 332236      0 0      144 0 402 1507 2 2 95 2
0 0      0 980672 59644 332240      0 0      0 0 285 1433 3 1 96 0
0 0      0 980672 59644 332240      0 0      0 0 376 1611 4 2 94 0
2 0      0 980144 59700 332184      0 0      0 188 1854 3733 10 13 67 10
1 1      0 980556 59716 332232      0 0      0 72 371 1603 3 2 87 9
0 0      0 980100 59748 332220      0 0      0 104 1091 2513 8 9 75 8
0 0      0 980052 59748 332240      0 0      0 0 292 1471 2 2 96 0
0 0      0 980052 59748 332240      0 0      0 0 627 1948 4 5 92 0
2 1      0 979820 59960 332188      0 0      12 600 2044 3175 14 14 41 32
0 0      0 979620 60096 332192      0 0      48 368 1941 3136 13 13 53 21
0 0      0 979448 60120 332280      0 0      88 0 1846 2776 9 10 75 6
0 0      0 978008 60184 332380      0 0      84 168 3008 4414 13 17 60 10
0 0      0 978072 60216 332444      0 0      68 52 2303 4348 11 13 64 12
0 1      0 978400 60272 332580      0 0      116 144 4028 6354 15 24 54 7
procs -----memory----- ---swap--- -----io----- -system-- -----cpu-----
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs us sy id wa
3 0      0 978104 60312 332928      0 0      316 80 3748 4970 15 21 57 7
0 0      0 977728 60336 333092      0 0      320 0 4986 5246 15 25 55 5
0 2      0 977192 60628 333324      0 0      136 712 3059 4356 15 20 35 30
1 0      0 976720 60728 333664      0 0      528 196 7396 8324 16 31 42 11
1 0      0 975116 60836 335932      0 0      2280 176 10590 16809 15 35 38 12
0 0      0 974136 60896 336348      0 0      392 124 4346 4715 14 20 54 11
3 1      0 971624 60936 339404      0 0      3100 96 11884 16436 16 38 38 8
3 0      0 969328 60960 340960      0 0      1492 64 8731 11006 17 31 36 15
1 0      0 965596 60960 345108      0 0      4084 0 14108 31799 15 42 41 2
0 0      0 965596 61008 345084      0 0      8 112 2672 3430 13 15 59 13
0 0      0 964240 61020 346620      0 0      1560 0 3782 6819 12 13 68 7
0 0      0 964156 61032 346664      0 0      4 28 469 783 9 3 73 16
0 0      0 964276 61072 346628      0 0      0 92 1019 2476 7 5 81 8
0 0      0 964028 61104 346668      0 0      0 116 900 2157 7 5 85 4
0 0      0 964100 61132 346652      0 0      4 72 3022 4181 11 19 64 6
3 0      0 964356 61140 346660      0 0      0 16 2283 3053 11 13 71 4
0 1      0 964380 61140 346768      0 0      72 0 3206 4719 14 19 63 4
0 0      0 964076 61156 346756      0 0      24 16 1109 1659 12 6 78 4
2 0      0 964060 61172 346756      0 0      0 28 2229 4117 11 15 69 5
0 0      0 964188 61188 346952      0 0      196 28 6004 8133 17 26 52 5
3 0      0 960964 61240 348228      0 0      1368 72 6571 9489 15 29 44 12
procs -----memory----- ---swap--- -----io----- -system-- -----cpu-----
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs us sy id wa
3 0      0 958992 61344 351804      0 0      3560 220 12811 23246 17 39 32 12
2 0      0 950400 61380 357656      0 0      5876 84 11487 17646 16 36 41 7
5 0      0 946972 61384 362056      0 0      4360 0 10515 16646 16 35 41 8
1 2      0 947352 61432 362932      0 0      856 84 4828 5623 17 26 45 12
3 0      0 946360 61448 363552      0 0      612 72 8861 20035 23 27 38 11
5 0      0 937784 61472 369072      0 0      5480 40 8887 54737 23 36 28 13
1 1      0 927124 61544 376276      0 0      6848 200 13523 60761 42 39 10 9
0 0      0 925580 61588 376368      0 0      44 128 1177 4414 44 11 42 4
0 0      0 926580 61588 376344      0 0      0 0 579 1016 10 4 86 0
0 0      0 926612 61588 376344      0 0      0 0 522 1957 12 3 85 0
0 0      0 926612 61588 376344      0 0      0 0 349 1600 5 2 94 0
1 0      0 926208 61588 376344      0 0      0 0 537 659 5 9 86 0
0 0      0 926364 61596 376244      0 0      0 588 346 1440 3 1 85 12
1 0      0 930828 61604 376340      0 0      0 72 648 1668 30 7 58 5
2 0      0 931464 61612 376312      0 0      20 80 704 1319 13 6 75 7
2 0      0 930720 61644 376420      0 0      32 268 1090 2190 13 5 75 7
0 0      0 930968 61652 376440      0 0      12 60 655 1435 4 1 83 12
0 0      0 930968 61652 376440      0 0      0 0 293 1297 2 1 97 0
0 0      0 930968 61652 376440      0 0      0 0 375 1675 3 2 95 0
0 0      0 930968 61652 376440      0 0      0 0 294 1402 3 1 96 0
1 0      0 930968 61652 376440      0 0      0 0 751 1859 7 2 91 0
procs -----memory----- ---swap--- -----io----- -system-- -----cpu-----
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs us sy id wa
2 0      0 930968 61660 376440      0 0      0 256 732 1554 2 1 84 13
0 0      0 930960 61660 376440      0 0      0 0 562 1374 4 1 95 0
0 0      0 930960 61660 376440      0 0      0 0 201 1233 0 0 100 0
0 0      0 930960 61668 376432      0 0      8 44 214 1255 1 1 93 5
0 0      0 930960 61668 376444      0 0      0 0 209 1285 0 1 99 0
0 0      0 930836 61668 376448      0 0      0 0 375 2466 5 2 93 0
1 0      0 930836 61668 376448      0 0      0 0 195 3828 1 1 98 0
0 0      0 930836 61668 376448      0 0      0 0 190 1269 1 1 99 0
0 0      0 930712 61676 376440      0 0      0 64 235 1289 1 0 91 8

```

```

1 0      0 930712 61676 376440 0 0 0 0 203 1198 1 0 99 0
0 0      0 930712 61684 376440 0 0 0 24 249 1363 2 1 92 5
0 0      0 930712 61684 376448 0 0 0 0 285 397 4 2 94 0
1 0      0 930712 61788 376352 0 0 0 228 616 2169 6 3 75 16
0 0      0 930712 61788 376448 0 0 0 0 221 1314 1 1 98 0
0 0      0 930712 61788 376448 0 0 0 0 248 1328 2 1 97 0
0 0      0 930712 61796 376448 0 0 0 52 260 1400 2 1 92 5
0 0      0 930712 61796 376448 0 0 0 0 208 1306 1 1 98 0
0 0      0 930712 61796 376448 0 0 0 0 179 1231 1 0 99 0
1 0      0 930712 61796 376452 0 0 0 0 194 1248 0 1 99 0
0 0      0 930712 61796 376452 0 0 0 0 211 1286 2 1 97 0
0 0      0 930712 61796 376452 0 0 0 44 294 1345 3 1 94 2
procs -----memory----- --swap-- --io-- --system-- --cpu--
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
4 0      0 930288 61900 376864 0 0 460 264 2151 4023 11 11 61 18
4 0      0 922136 61908 383388 0 0 6496 0 13028 23771 16 33 45 6
3 5      0 913696 61916 392916 0 0 9552 32 10033 13785 15 38 36 11
1 0      0 903240 61928 402156 0 0 9220 0 11890 18864 17 36 33 13
4 0      0 893708 61936 411680 0 0 9564 0 12862 28426 19 38 34 9
3 0      0 884268 61948 420936 0 0 9220 0 14322 29011 15 42 34 9
3 1      0 872188 61948 432200 0 0 11268 0 19469 46374 17 47 34 3
0 0      0 863728 61948 442312 0 0 10088 0 17927 34379 19 43 35 4
1 0      0 862332 61976 444604 0 0 2340 16 8238 8343 21 36 36 7
3 1      0 861696 61992 445396 0 0 920 36 6203 7843 17 31 46 6
2 0      0 849360 62052 455568 0 0 10152 0 19430 36414 14 47 32 6
0 0      0 846080 62068 460708 0 0 5180 0 10886 18523 17 38 39 6
1 2      0 845932 62088 460780 0 0 116 20 2388 3324 12 12 62 14
0 0      0 844684 62164 461620 0 0 844 104 3716 5946 13 19 54 13
2 0      0 844324 62224 462056 0 0 464 144 2791 3445 14 14 64 9
1 0      0 844340 62288 462260 0 0 172 44 5322 6413 13 34 48 5
0 0      0 842896 62288 463272 0 0 1120 0 4685 5274 12 26 60 3
3 1      0 840204 62288 466400 0 0 3052 0 8137 10757 15 35 41 10
0 0      0 838528 62304 468120 0 0 1664 36 4184 4781 16 17 51 15
1 0      0 836696 62380 469768 0 0 1640 24 3378 5067 14 18 57 11
1 0      0 835652 62464 470896 0 0 1340 72 3434 5265 15 21 53 12
procs -----memory----- --swap-- --io-- --system-- --cpu--
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1 0      0 834224 62484 472500 0 0 1444 56 5659 6417 15 31 48 6
3 2      0 832968 62516 473860 0 0 1424 84 3822 4736 13 24 54 9
2 0      0 832592 62524 474060 0 0 180 52 3224 4520 16 18 61 5
2 0      0 831300 62536 474112 0 0 12 24 2961 5346 17 18 61 4
4 0      0 829560 62544 474112 0 0 0 24 3476 5225 17 21 59 3
0 0      0 830468 62548 474216 0 0 152 0 4211 5946 20 23 50 7
0 0      0 830476 62552 474268 0 0 4 0 1430 3270 7 8 81 4
1 0      0 825700 62552 474268 0 0 0 0 668 2649 19 7 74 0
3 0      0 824520 62552 474108 0 0 0 0 1231 3317 58 13 30 0
0 0      0 824220 62552 474056 0 0 0 0 933 2883 16 7 77 0
1 0      0 824964 62560 474032 0 0 0 24 352 1510 4 2 89 6
0 0      0 824468 62560 474032 0 0 0 0 434 732 9 3 88 0
0 0      0 825336 62560 474032 0 0 0 0 273 1390 3 1 96 0
0 0      0 825368 62560 474032 0 0 0 0 350 1539 3 3 94 0
0 0      0 825492 62560 473932 0 0 0 0 332 1570 3 1 96 0
0 1      0 825040 62564 473932 0 0 0 128 507 1711 18 3 79 0
0 0      0 829748 62576 474008 0 0 0 56 1119 3133 25 5 65 5
0 0      0 830368 62616 474004 0 0 16 328 768 649 6 2 79 13
0 0      0 830368 62616 474028 0 0 0 0 397 1512 3 1 96 0
0 0      0 830368 62616 474028 0 0 0 0 351 1487 4 2 95 0
0 0      0 830368 62616 474028 0 0 0 0 260 1382 2 1 97 0
procs -----memory----- --swap-- --io-- --system-- --cpu--
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1 0      0 830120 62616 474028 0 0 0 0 253 1463 2 1 97 0
0 0      0 830120 62624 474024 0 0 0 204 257 1357 1 1 88 10
1 0      0 830120 62624 474032 0 0 0 0 312 470 5 2 94 0
0 0      0 830120 62624 474032 0 0 0 0 213 1305 1 0 99 0
0 0      0 830120 62624 474032 0 0 0 0 201 1242 1 1 98 0
2 0      0 830120 62624 474032 0 0 0 0 220 1302 2 0 98 0
0 0      0 830120 62640 474016 0 0 0 92 246 1416 2 1 89 9
0 0      0 830120 62640 474036 0 0 0 0 260 1351 3 1 97 0
1 0      0 829624 62640 474036 0 0 0 0 777 1761 10 4 86 0
0 0      0 830120 62640 474036 0 0 0 0 790 1628 6 2 93 0
0 0      0 830120 62640 474036 0 0 0 0 382 1319 1 0 99 0
1 1      0 830120 62648 474028 0 0 0 44 266 1334 2 1 92 5
0 0      0 830120 62648 474036 0 0 0 0 253 1397 2 1 94 4
0 0      0 830120 62648 474036 0 0 0 0 173 1216 1 1 99 0
0 0      0 830120 62648 474036 0 0 0 0 187 1224 0 1 99 0
1 0      0 830120 62648 474036 0 0 0 0 303 1589 7 2 91 0
0 0      0 830128 62648 474036 0 0 0 0 205 1246 2 1 98 0

```

```

1 0      0 830112 62656 474036    0    0    0    16 340 1428 3 1 90 6
0 0      0 827888 62656 476936    0    0 2900    0 3426 3997 12 16 64 8
4 0      0 813680 62656 489332    0    0 12432    0 21676 48088 17 48 32 3
6 0      0 802036 62656 500852    0    0 11520    0 22414 52763 20 47 32 0
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free   buff  cache  si  so   bi   bo   in  cs us sy id wa
3 0      0 798308 62680 506348    0    0 5516   20 13897 24287 17 36 44 3
2 1      0 792396 62720 512380    0    0 6072   20 11742 28068 18 37 39 7
1 0      0 791152 62828 513252    0    0 1016   36 4455 5569 15 21 52 13
1 0      0 789540 62848 514456    0    0 1168   24 2468 3036 10 16 65 10
0 0      0 789344 62868 514836    0    0  416    0  837 2167 5 4 87 4
0 1      0 789236 62888 514884    0    0  76    0  743 1233 10 4 76 9
0 1      0 787084 62972 517240    0    0 1888  236 3217 3425 14 16 54 16
0 0      0 786476 63040 517724    0    0  416  128 1165 1701 9 6 75 11
4 0      0 786152 63088 517916    0    0  0    480 1650 2181 11 7 76 7
0 0      0 786152 63088 517876    0    0  0    0  331  415 3 2 95 0
4 0      0 781676 63088 517992    0    0  0    0 1117 2692 21 10 70 0
0 0      0 779468 63088 518064    0    0  0    0  890 3247 57 9 34 0
0 0      0 779328 63088 518012    0    0  0    0  818 1582 12 6 82 0
0 0      0 779948 63088 517988    0    0  0    72 340  518 5 3 93 0
0 0      0 779576 63096 517988    0    0  0    24 363  599 5 3 88 5
0 0      0 780568 63096 517988    0    0  0    0  294 1407 3 1 97 0
0 0      0 780596 63096 517988    0    0  64    0  630 1970 6 2 89 3
0 0      0 780688 63096 517952    0    0  0    0  510  851 5 2 93 0
1 1      0 780192 63104 518048    0    0  0    60 732  997 24 4 66 5
0 1      0 784672 63112 518024    0    0  24   80 1039 1793 19 6 72 4
0 0      0 785648 63144 518056    0    0  0   260 876  931 8 2 81 8
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free   buff  cache  si  so   bi   bo   in  cs us sy id wa
1 0      0 785656 63144 518056    0    0  0    0 755  605 7 3 90 0
0 0      0 785656 63144 518060    0    0  0    0 266  366 1 2 97 0
0 0      0 785656 63144 518060    0    0  0    0 284  411 5 1 95 0
0 0      0 785656 63144 518060    0    0  0    0 226  304 1 1 98 0
0 0      0 785656 63152 518052    0    0  0  196 232 1358 1 1 89 10
0 0      0 785656 63152 518060    0    0  0    0 283  377 3 1 97 0
0 0      0 785656 63152 518060    0    0  0    0 228  305 0 1 99 0
0 0      0 785656 63152 518064    0    0  0    0 206 1150 1 1 98 0
0 0      0 785656 63152 518064    0    0  0    0 197  214 1 0 99 0
1 0      0 800288 63160 518056    0    0  0    36 271  359 4 2 92 2
0 0      0 800296 63160 518064    0    0  0    0 236  311 2 1 98 0
0 0      0 800296 63160 518064    0    0  0    0 206  262 1 1 99 0
1 0      0 800296 63160 518064    0    0  0    0 215  277 1 1 99 0
0 0      0 800296 63160 518064    0    0  0    0 190 1177 1 1 99 0
0 0      0 799800 63172 518052    0    0  0    84 299  394 5 1 86 9
1 1      0 799808 63180 518056    0    0  0    36 241 1319 1 0 95 5
0 0      0 799684 63264 518004    0    0  4   264 562 1936 2 2 75 22
0 0      0 799708 63288 518192    0    0 192   72 597  795 2 2 92 5
0 0      0 799708 63288 518192    0    0  0    0 197  220 0 1 99 0
0 0      0 799708 63288 518192    0    0  0    0 201  289 1 1 98 0
0 1      0 799708 63368 518112    0    0  0   336 475 4243 1 3 81 15
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free   buff  cache  si  so   bi   bo   in  cs us sy id wa
3 0      0 799708 63368 518184    0    0  0    0 217 1983 2 1 97 0
0 0      0 799708 63376 518176    0    0  20   60 791 2467 1 3 90 6
0 1      0 799212 63388 518192    0    0  32    0 1677 3253 2 8 82 8
1 0      0 802104 63388 518212    0    0  0    0 476  946 11 3 82 4

```

ANEXO K - Teste de Processador – Comando vmstat - Servidor DRBL  
Etapas de 2 a 4 – Thin Client 01

```

procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
4  1    0 1117116 52952 222820  0  0 1540  30 338 1146 17  7 62 14
1  0    0 1116920 52956 222956  0  0  148  0 814  868 51  3 44  2
2  0    0 1138004 53284 223976  0  0 1376  0 774 2364 24  6 37 33
1  2    0 1122084 53364 229160  0  0 5248  0 935 2984 42  8 14 36
2  3    0 1108432 53416 233572  0  0 4436  0 3011 5707 52 19  3 26
7  0    0 1097336 53732 238636  0  0 5368  0 5616 9439 40 25  5 31
2  1    0 1098516 54152 240728  0  0 2556 44 4725 8662 50 17 11 22
3  1    0 1082796 54212 246512  0  0 5768  0 5800 9327 46 28 11 15
1  1    0 1069328 54224 259888  0  0 13376 52 5746 8765 21 26 16 37
2  1    0 1058532 54224 270036  0  0 10112  0 5600 9042 22 23 27 28
0  1    0 1053680 54224 274184  0  0 4056  0 5172 7481 43 20 23 13
3  0    0 1046108 54224 278840  0  0 4708  0 6380 8003 46 18 30  6
3  0    0 1043372 54224 281568  0  0 2752  0 3948 6366 57 16 25  2
1  0    0 1042380 54232 282412  0  0  824 52 2706 4640 22  8 69  2
0  0    0 1042140 54232 282420  0  0  0 0 311 1523  3  2 95  0
0  0    0 1042132 54232 282420  0  0  0 0 292 1446  2  1 97  0
1  0    0 1042140 54232 282416  0  0  0 0 260 1422  3  1 96  0
1  0    0 1042140 54232 282416  0  0  0 0 486 1561  5  2 93  0
0  1    0 1038300 54848 285308  0  0 3608 52 3786 4621  7 13 59 22
2  0    0 1034988 55004 287780  0  0 2580 44 4577 7475 14 21 47 18
0  0    0 1034944 55308 287928  0  0  408 204 2760 4646 11 16 52 21
procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2  1    0 1032200 55608 290052  0  0 2300  0 4593 6812 16 25 41 19
1  0    0 1029736 55704 292380  0  0 2440 292 7229 7202 18 30 22 30
3  0    0 1029160 55912 292728  0  0  232 596 2451 3886  9 17 46 27
0  5    0 1026776 56144 294724  0  0 2364 188 7765 8100 16 35 13 36
4  4    0 1020120 56320 299940  0  0 5292 152 11478 15113 15 38 11 37
3  0    0 1019300 56512 302068  0  0 2144 324 8191 10526 14 35 24 28
1  0    0 1017128 56688 303908  0  0 1760 412 6969 6708 19 26 32 23
0  0    0 1015528 56808 305412  0  0 1392 488 6303 6270 13 27 41 19
1  1    0 1014040 56868 306124  0  0  724 276 3232 4340 13 18 54 15
2  1    0 1013384 56948 307660  0  0 1420 224 7550 7841 15 37 31 17
1  1    0 1010596 57024 310156  0  0 2552  76 8638 8936 15 36 35 14
4  5    0 1006980 57148 313332  0  0 3308 144 12515 18638 14 41 27 17
1  1    0 1005416 57348 314260  0  0 1104 476 9876 11028 17 37 17 29
2  0    0 1002652 57384 317036  0  0 2552 176 6811 8117 15 26 45 14
3  1    0 1002272 57428 317536  0  0  636  64 5673 6064 15 23 47 15
6  1    0 1000476 57508 317888  0  0  344  92 6586 7945 14 35 43  8
4  0    0 998960 57676 318308  0  0 452 372 5579 9964 12 30 39 19
3  0    0 1000068 57680 319336  0  0  952  0 12605 20750 15 39 43  3
1  1    0 998360 57900 319696  0  0  524 400 6298 7060 16 30 27 27
2  2    0 997568 58164 319844  0  0  16 832 5262 6200 15 25 28 32
3  0    0 988848 58216 327260  0  0 7408 148 21625 38383 17 46 23 14
procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1  2    0 989056 58420 328484  0  0 1408 560 8077 12327 14 34 21 30
3  0    0 988524 58640 329152  0  0 480 768 6256 6692 19 26 24 31
2  1    0 987700 58888 329132  0  0 144 936 7461 7773 17 33 25 25
2  2    0 986252 59036 331172  0  0 2112 388 11666 10695 18 38 27 16
1  0    0 984604 59176 332212  0  0 1016 192 7802 7320 15 34 35 16
1  1    0 983784 59340 332364  0  0  20 264 2515 3875 11 15 50 23
0  0    0 983476 59372 332572  0  0 276  72 4528 4624 16 17 63  5
0  0    0 982808 59624 332584  0  0 288  0 589 825  5  4 85  7
1  0    0 982800 59624 332644  0  0  0 0 289 1440  2  3 95  0
0  1    0 982800 59640 332628  0  0  0 44 1550 3107 10  9 78  4
0  0    0 982800 59688 332604  0  0  0 172 810 2424  6  5 81  9
2  0    0 982800 59704 332624  0  0  0 68 419 1766  3  2 87  8
0  0    0 982304 59728 332632  0  0  0 92 1014 1600  7  6 79  8
0  0    0 982304 59728 332640  0  0  0 0 333 1551  3  3 94  0
0  1    0 982180 59884 332616  0  0 132 116 1134 2812  5  7 77 11
2  0    0 976048 60064 333724  0  0 1096 452 1957 3904 28 14 20 39
1  0    0 979888 60160 333756  0  0 152 208 3466 5797 20 20 46 14
1  1    0 979204 60244 333780  0  0  92 160 4238 6229 15 21 54 10
3  0    0 978548 60352 333884  0  0 132 296 5304 6964 15 31 42 12
1  1    0 976876 60368 334632  0  0 620  0 6263 7174 17 25 49  9
0  2    0 977436 60668 334780  0  0 208 600 4136 6341 15 23 26 36
procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
3  0    0 973936 60792 337332  0  0 2688 244 13667 19961 17 42 28 12

```

```

4 1      0 972176 60832 340404 0 0 3084 96 14782 19234 16 43 34 7
2 0      0 964312 60924 345388 0 0 4976 240 16917 35393 23 44 19 14
2 0      0 963760 61024 346092 0 0 756 148 12387 15124 17 41 34 8
0 0      0 963252 61052 348208 0 0 2036 20 8013 7904 15 25 54 5
0 0      0 963572 61100 348240 0 0 36 144 1829 2744 7 10 71 12
2 0      0 963304 61116 348272 0 0 0 64 4033 5820 21 23 54 3
1 0      0 956076 61140 348368 0 0 148 48 2013 3344 27 16 32 24
1 0      0 956808 61196 348472 0 0 76 140 3094 4307 28 19 47 5
2 0      0 955608 61224 349852 0 0 1388 40 14564 29193 17 38 35 10
3 0      0 955784 61296 350016 0 0 256 64 9906 11332 15 40 38 7
0 0      0 955760 61296 350112 0 0 0 0 3299 5004 12 16 72 0
0 0      0 954568 61296 350112 0 0 0 0 363 1669 11 2 87 0
1 0      0 954552 61296 350112 0 0 0 0 197 1235 1 1 99 0
0 0      0 955668 61312 350096 0 0 0 36 429 620 3 2 89 6
4 0      0 952388 61352 351772 0 0 1748 48 5812 18898 14 23 55 8
6 0      0 937308 61420 360536 0 0 8304 120 17645 73051 39 47 9 6
1 0      0 934452 61436 364440 0 0 3888 0 7964 43698 54 25 21 1
0 0      0 933824 61436 364416 0 0 0 0 768 1504 15 4 81 0
1 0      0 934312 61436 364416 0 0 0 0 284 1426 1 1 98 0
2 0      0 931996 61484 365868 0 0 1540 128 5380 9440 15 19 52 14
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2 0      0 932468 61504 366096 0 0 116 20 1761 3098 11 7 76 6
0 0      0 932576 61504 366084 0 0 0 0 302 1506 4 1 95 0
0 0      0 932708 61504 365988 0 0 0 0 394 1754 8 2 90 0
0 0      0 932708 61504 365988 0 0 0 0 406 1492 4 2 94 0
0 0      0 932708 61504 365988 0 0 0 0 348 1649 3 2 96 0
1 0      0 932628 61512 365984 0 0 0 40 340 1626 6 3 87 5
0 0      0 937460 61520 366008 0 0 0 56 1102 2048 36 9 53 3
0 0      0 936640 61560 366020 0 0 16 328 881 2707 10 3 74 13
0 0      0 937664 61560 366092 0 0 0 0 648 1546 7 1 92 0
0 0      0 937788 61560 366092 0 0 0 0 356 1545 3 3 95 0
0 0      0 937788 61560 366096 0 0 0 0 368 1493 5 2 94 0
0 0      0 937788 61568 366092 0 0 0 56 444 1736 3 1 92 5
1 0      0 937788 61568 366100 0 0 0 0 575 872 5 2 93 0
0 0      0 937788 61568 366100 0 0 0 0 695 1551 2 2 96 0
0 0      0 937292 61568 366100 0 0 0 0 828 1863 10 4 87 0
1 0      0 937292 61568 366100 0 0 0 0 372 1292 1 1 98 0
0 0      0 937292 61568 366100 0 0 0 708 297 1222 1 1 99 0
0 0      0 937292 61576 366100 0 0 0 40 173 1204 0 1 82 18
1 0      0 937292 61576 366100 0 0 0 0 195 1239 1 1 98 0
1 0      0 937344 61576 366100 0 0 0 0 557 808 7 9 84 0
0 0      0 937368 61576 366100 0 0 0 0 232 1339 1 1 98 0
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
3 1      0 937368 61640 366036 0 0 0 164 410 1661 3 3 80 15
0 0      0 937368 61688 366056 0 0 0 104 399 1636 4 3 86 8
0 0      0 937368 61688 366100 0 0 0 0 182 1240 0 0 100 0
1 0      0 937368 61688 366100 0 0 0 0 213 1311 1 1 98 0
0 0      0 937368 61688 366100 0 0 0 0 184 1252 1 1 99 0
0 0      0 937368 61688 366100 0 0 0 0 193 1216 0 0 100 0
1 0      0 937368 61696 366096 0 0 0 76 214 1210 1 0 92 6
0 0      0 937336 61696 366100 0 0 0 0 179 1228 1 0 99 0
0 0      0 937336 61696 366100 0 0 0 0 201 1255 1 1 99 0
1 0      0 937336 61696 366100 0 0 0 0 179 1220 1 1 99 0
0 0      0 937336 61696 366100 0 0 0 0 165 1215 0 1 99 0
0 0      0 937336 61696 366100 0 0 0 36 167 1210 1 0 99 0
1 0      0 937336 61704 366100 0 0 0 16 194 1261 1 1 94 5
0 0      0 937336 61704 366100 0 0 0 0 258 1364 2 1 98 0
0 0      0 937336 61704 366100 0 0 0 0 198 1258 1 1 98 0
1 0      0 937416 61704 366100 0 0 0 0 211 1285 2 1 98 0
0 0      0 937432 61704 366100 0 0 0 0 191 1247 0 0 100 0
0 0      0 937432 61704 366100 0 0 0 4 244 1329 1 1 99 0
0 1      0 934992 61752 368860 0 0 2808 28 10131 13381 15 36 35 14
1 0      0 925480 61764 378196 0 0 9400 0 17849 28759 18 39 33 10
1 0      0 916444 61764 387012 0 0 8812 0 11275 10349 18 35 36 11
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
4 1      0 904536 61784 397544 0 0 10612 0 18607 32485 19 40 34 7
3 0      0 893888 61784 407448 0 0 9756 0 17270 33391 17 45 30 8
3 0      0 882836 61784 418156 0 0 10804 0 21894 43761 16 46 35 3
4 0      0 872608 61792 428564 0 0 10592 16 21012 41195 17 40 35 8
6 0      0 868648 61828 434220 0 0 5480 20 14549 20510 17 41 33 8
4 0      0 860812 61852 440248 0 0 6044 52 18770 32605 16 48 29 7
3 2      0 855184 61912 447332 0 0 7168 0 17802 34033 18 47 32 3
3 6      0 842888 61916 459716 0 0 12336 0 22642 45793 19 45 32 4
2 1      0 840436 61956 462640 0 0 2960 28 14272 18658 16 49 29 6

```

```

3 0      0 837784 62068 464548 0 0 1984 140 8841 9958 17 42 25 16
0 0      0 833824 62124 468608 0 0 4040 92 10523 13225 15 44 30 11
2 0      0 829388 62224 472632 0 0 4160 76 10541 10828 15 47 26 13
0 1      0 827164 62340 475372 0 0 2764 140 8321 8319 19 34 29 18
0 0      0 825856 62364 475732 0 0 252 56 1646 2108 7 10 73 10
0 0      0 825792 62388 475740 0 0 0 84 327 1506 3 2 89 7
0 0      0 825512 62388 475764 0 0 0 0 333 1611 9 2 90 0
2 0      0 819944 62388 475664 0 0 0 0 1151 1982 32 12 56 0
1 0      0 819084 62388 475608 0 0 0 0 1014 3325 47 9 44 0
0 0      0 819736 62388 475556 0 0 0 0 620 2146 10 4 86 0
0 0      0 819736 62396 475524 0 0 0 120 363 1588 5 3 86 7
0 0      0 819364 62396 475532 0 0 0 0 431 1798 8 2 90 0
procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2 0      0 819736 62396 475532 0 0 0 0 322 1524 3 1 96 0
2 0      0 819736 62396 475532 0 0 0 0 513 1985 6 3 91 0
0 0      0 824696 62404 475524 0 0 0 60 735 1781 33 9 54 5
0 0      0 824736 62444 475468 0 0 0 332 958 1028 10 3 75 12
1 0      0 824596 62444 475508 0 0 0 0 816 781 12 4 85 0
0 0      0 824480 62444 475508 0 0 0 0 510 1467 3 1 96 0
0 0      0 825720 62444 475512 0 0 0 0 318 1538 3 2 96 0
0 0      0 825720 62444 475512 0 0 0 0 266 1328 3 0 97 0
0 1      0 825720 62452 475504 0 0 0 216 368 1442 3 1 88 9
0 0      0 825720 62452 475512 0 0 0 0 706 1260 2 2 95 2
0 0      0 825720 62452 475512 0 0 0 0 783 2104 10 2 89 0
0 0      0 825232 62452 475512 0 0 0 0 619 5579 3 1 96 0
0 0      0 825232 62452 475512 0 0 0 0 193 2805 0 0 100 0
0 0      0 825232 62452 475512 0 0 0 0 202 1271 0 1 99 0
0 0      0 825232 62460 475512 0 0 0 60 198 1244 1 0 92 7
0 0      0 825232 62460 475512 0 0 0 0 185 1224 1 0 99 0
0 0      0 825728 62460 475512 0 0 0 0 334 1511 3 1 96 0
0 0      0 825728 62460 475512 0 0 0 0 238 1340 3 1 96 0
0 1      0 825728 62468 475504 0 0 0 24 259 1391 2 1 93 5
0 0      0 825696 62468 475512 0 0 0 0 218 1308 2 0 98 0
0 0      0 825696 62468 475512 0 0 0 0 193 1224 1 1 99 0
procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1 0      0 825696 62468 475512 0 0 0 0 340 1558 2 2 97 0
1 0      0 825712 62468 475512 0 0 0 0 228 1359 2 1 97 0
0 1      0 825696 62476 475504 0 0 0 24 190 1238 0 1 94 5
0 0      0 825696 62476 475512 0 0 0 0 211 1287 2 1 97 1
1 0      0 825696 62476 475512 0 0 0 0 176 1215 0 0 100 0
0 0      0 825696 62476 475512 0 0 0 0 210 1256 1 0 99 0
0 0      0 825696 62476 475512 0 0 0 0 183 1201 0 1 99 0
1 0      0 825696 62476 475512 0 0 0 0 206 1304 1 1 98 0
0 0      0 825696 62484 475512 0 0 0 36 240 1339 1 1 91 7
0 0      0 825680 62484 475512 0 0 0 0 219 1283 1 1 98 0
0 0      0 825696 62484 475512 0 0 0 0 247 1384 1 1 98 0
0 0      0 825696 62484 475512 0 0 0 0 226 1311 1 1 98 0
0 0      0 825696 62484 475512 0 0 0 0 165 1195 0 0 100 0
0 0      0 825696 62484 475512 0 0 0 4 200 1266 1 0 99 0
0 0      0 825696 62492 475512 0 0 0 16 229 1306 1 1 94 5
4 0      0 816252 62492 483312 0 0 7908 0 12468 19614 15 31 43 11
1 0      0 804760 62492 494988 0 0 11520 0 25336 49376 18 45 36 1
0 0      0 792912 62492 506508 0 0 11520 0 25890 62564 15 52 33 0
2 0      0 786404 62532 512888 0 0 6572 32 16183 24631 16 43 35 6
3 0      0 786388 62688 514420 0 0 1648 120 7993 10956 15 31 42 13
0 1      0 784740 62740 516276 0 0 1900 52 3619 4449 12 18 54 15
procs -----memory----- --swap-- -----io----- --system-- -----cpu-----
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1 1      0 781532 62872 519180 0 0 2288 324 3849 4014 13 16 45 25
0 0      0 781448 62908 519172 0 0 64 44 787 1186 5 3 87 5
4 0      0 781056 62956 519396 0 0 0 480 1316 2661 7 3 79 12
0 0      0 781056 62956 519408 0 0 0 0 501 1244 6 3 91 0
1 0      0 775236 62956 519572 0 0 0 0 1398 3106 33 11 56 0
2 0      0 773644 62956 519544 0 0 0 0 891 2925 49 9 42 0
0 0      0 773816 62956 519492 0 0 0 0 667 1901 11 5 84 0
0 0      0 774824 62956 519468 0 0 0 68 330 1537 6 1 93 0
0 0      0 774824 62964 519468 0 0 0 32 378 626 5 3 88 5
0 0      0 774824 62964 519468 0 0 0 0 390 1722 6 2 92 0
0 0      0 774700 62964 519468 0 0 64 0 812 2068 5 4 87 4
2 0      0 780328 62972 519524 0 0 0 60 641 2610 31 7 57 5
1 1      0 779196 62996 519484 0 0 0 200 969 3375 18 4 71 7
0 0      0 779816 63012 519484 0 0 0 136 713 1585 4 2 90 5
0 0      0 779816 63012 519508 0 0 0 0 381 1496 3 1 96 0
1 0      0 779816 63012 519512 0 0 0 0 245 1360 1 1 98 0
0 0      0 779816 63012 519512 0 0 0 0 244 1384 1 1 98 0

```

0	0	0	779816	63012	519512	0	0	0	0	202	1313	1	1	99	0
1	0	0	779848	63020	519508	0	0	0	188	285	1421	1	1	88	11
0	0	0	779864	63020	519512	0	0	0	0	682	1595	2	2	97	0
0	0	0	779864	63020	519512	0	0	0	0	664	1442	4	1	95	0

ANEXO L - Teste de Processador – Comando vmstat - Servidor DRBL  
Etapas de 2 a 4 – Thin Client 02

```

procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
4  1  0  1117116  52952  222820  0  0  1540  30  338  1146  17  7  62  14
1  0  0  1116920  52956  222956  0  0  148  0  814  868  51  3  44  2
2  0  0  1138004  53284  223976  0  0  1376  0  774  2364  24  6  37  33
1  2  0  1122084  53364  229160  0  0  5248  0  935  2984  42  8  14  36
2  3  0  1108432  53416  233572  0  0  4436  0  3011  5707  52  19  3  26
7  0  0  1097336  53732  238636  0  0  5368  0  5616  9439  40  25  5  31
2  1  0  1098516  54152  240728  0  0  2556  44  4725  8662  50  17  11  22
3  1  0  1082796  54212  246512  0  0  5768  0  5800  9327  46  28  11  15
1  1  0  1069328  54224  259888  0  0  13376  52  5746  8765  21  26  16  37
2  1  0  1058532  54224  270036  0  0  10112  0  5600  9042  22  23  27  28
0  1  0  1053680  54224  274184  0  0  4056  0  5172  7481  43  20  23  13
3  0  0  1046108  54224  278840  0  0  4708  0  6380  8003  46  18  30  6
3  0  0  1043372  54224  281568  0  0  2752  0  3948  6366  57  16  25  2
1  0  0  1042380  54232  282412  0  0  824  52  2706  4640  22  8  69  2
0  0  0  1042140  54232  282420  0  0  0  0  311  1523  3  2  95  0
0  0  0  1042132  54232  282420  0  0  0  0  292  1446  2  1  97  0
1  0  0  1042140  54232  282416  0  0  0  0  260  1422  3  1  96  0
1  0  0  1042140  54232  282416  0  0  0  0  486  1561  5  2  93  0
0  1  0  1038300  54848  285308  0  0  3608  52  3786  4621  7  13  59  22
2  0  0  1034988  55004  287780  0  0  2580  44  4577  7475  14  21  47  18
0  0  0  1034944  55308  287928  0  0  408  204  2760  4646  11  16  52  21
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2  1  0  1032200  55608  290052  0  0  2300  0  4593  6812  16  25  41  19
1  0  0  1029736  55704  292380  0  0  2440  292  7229  7202  18  30  22  30
3  0  0  1029160  55912  292728  0  0  232  596  2451  3886  9  17  46  27
0  5  0  1026776  56144  294724  0  0  2364  188  7765  8100  16  35  13  36
4  4  0  1020120  56320  299940  0  0  5292  152  11478  15113  15  38  11  37
3  0  0  1019300  56512  302068  0  0  2144  324  8191  10526  14  35  24  28
1  0  0  1017128  56688  303908  0  0  1760  412  6969  6708  19  26  32  23
0  0  0  1015528  56808  305412  0  0  1392  488  6303  6270  13  27  41  19
1  1  0  1014040  56868  306124  0  0  724  276  3232  4340  13  18  54  15
2  1  0  1013384  56948  307660  0  0  1420  224  7550  7841  15  37  31  17
1  1  0  1010596  57024  310156  0  0  2552  76  8638  8936  15  36  35  14
4  5  0  1006980  57148  313332  0  0  3308  144  12515  18638  14  41  27  17
1  1  0  1005416  57348  314260  0  0  1104  476  9876  11028  17  37  17  29
2  0  0  1002652  57384  317036  0  0  2552  176  6811  8117  15  26  45  14
3  1  0  1002272  57428  317536  0  0  636  64  5673  6064  15  23  47  15
6  1  0  1000476  57508  317888  0  0  344  92  6586  7945  14  35  43  8
4  0  0  998960  57676  318308  0  0  452  372  5579  9964  12  30  39  19
3  0  0  1000068  57680  319336  0  0  952  0  12605  20750  15  39  43  3
1  1  0  998360  57900  319696  0  0  524  400  6298  7060  16  30  27  27
2  2  0  997568  58164  319844  0  0  16  832  5262  6200  15  25  28  32
3  0  0  988848  58216  327260  0  0  7408  148  21625  38383  17  46  23  14
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1  2  0  989056  58420  328484  0  0  1408  560  8077  12327  14  34  21  30
3  0  0  988524  58640  329152  0  0  480  768  6256  6692  19  26  24  31
2  1  0  987700  58888  329132  0  0  144  936  7461  7773  17  33  25  25
2  2  0  986252  59036  331172  0  0  2112  388  11666  10695  18  38  27  16
1  0  0  984604  59176  332212  0  0  1016  192  7802  7320  15  34  35  16
1  1  0  983784  59340  332364  0  0  20  264  2515  3875  11  15  50  23
0  0  0  983476  59372  332572  0  0  276  72  4528  4624  16  17  63  5
0  0  0  982808  59624  332584  0  0  288  0  589  825  5  4  85  7
1  0  0  982800  59624  332644  0  0  0  0  289  1440  2  3  95  0
0  1  0  982800  59640  332628  0  0  0  44  1550  3107  10  9  78  4
0  0  0  982800  59688  332604  0  0  0  172  810  2424  6  5  81  9
2  0  0  982800  59704  332624  0  0  0  68  419  1766  3  2  87  8
0  0  0  982304  59728  332632  0  0  0  92  1014  1600  7  6  79  8
0  0  0  982304  59728  332640  0  0  0  0  333  1551  3  3  94  0
0  1  0  982180  59884  332616  0  0  132  116  1134  2812  5  7  77  11
2  0  0  976048  60064  333724  0  0  1096  452  1957  3904  28  14  20  39
1  0  0  979888  60160  333756  0  0  152  208  3466  5797  20  20  46  14
1  1  0  979204  60244  333780  0  0  92  160  4238  6229  15  21  54  10
3  0  0  978548  60352  333884  0  0  132  296  5304  6964  15  31  42  12
1  1  0  976876  60368  334632  0  0  620  0  6263  7174  17  25  49  9
0  2  0  977436  60668  334780  0  0  208  600  4136  6341  15  23  26  36
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r  b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
3  0  0  973936  60792  337332  0  0  2688  244  13667  19961  17  42  28  12

```

```

4 1      0 972176 60832 340404 0 0 3084 96 14782 19234 16 43 34 7
2 0      0 964312 60924 345388 0 0 4976 240 16917 35393 23 44 19 14
2 0      0 963760 61024 346092 0 0 756 148 12387 15124 17 41 34 8
0 0      0 963252 61052 348208 0 0 2036 20 8013 7904 15 25 54 5
0 0      0 963572 61100 348240 0 0 36 144 1829 2744 7 10 71 12
2 0      0 963304 61116 348272 0 0 0 64 4033 5820 21 23 54 3
1 0      0 956076 61140 348368 0 0 148 48 2013 3344 27 16 32 24
1 0      0 956808 61196 348472 0 0 76 140 3094 4307 28 19 47 5
2 0      0 955608 61224 349852 0 0 1388 40 14564 29193 17 38 35 10
3 0      0 955784 61296 350016 0 0 256 64 9906 11332 15 40 38 7
0 0      0 955760 61296 350112 0 0 0 0 3299 5004 12 16 72 0
0 0      0 954568 61296 350112 0 0 0 0 363 1669 11 2 87 0
1 0      0 954552 61296 350112 0 0 0 0 197 1235 1 1 99 0
0 0      0 955668 61312 350096 0 0 0 36 429 620 3 2 89 6
4 0      0 952388 61352 351772 0 0 1748 48 5812 18898 14 23 55 8
6 0      0 937308 61420 360536 0 0 8304 120 17645 73051 39 47 9 6
1 0      0 934452 61436 364440 0 0 3888 0 7964 43698 54 25 21 1
0 0      0 933824 61436 364416 0 0 0 0 768 1504 15 4 81 0
1 0      0 934312 61436 364416 0 0 0 0 284 1426 1 1 98 0
2 0      0 931996 61484 365868 0 0 1540 128 5380 9440 15 19 52 14
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2 0      0 932468 61504 366096 0 0 116 20 1761 3098 11 7 76 6
0 0      0 932576 61504 366084 0 0 0 0 302 1506 4 1 95 0
0 0      0 932708 61504 365988 0 0 0 0 394 1754 8 2 90 0
0 0      0 932708 61504 365988 0 0 0 0 406 1492 4 2 94 0
0 0      0 932708 61504 365988 0 0 0 0 348 1649 3 2 96 0
1 0      0 932628 61512 365984 0 0 0 40 340 1626 6 3 87 5
0 0      0 937460 61520 366008 0 0 0 56 1102 2048 36 9 53 3
0 0      0 936640 61560 366020 0 0 16 328 881 2707 10 3 74 13
0 0      0 937664 61560 366092 0 0 0 0 648 1546 7 1 92 0
0 0      0 937788 61560 366092 0 0 0 0 356 1545 3 3 95 0
0 0      0 937788 61560 366096 0 0 0 0 368 1493 5 2 94 0
0 0      0 937788 61568 366092 0 0 0 56 444 1736 3 1 92 5
1 0      0 937788 61568 366100 0 0 0 0 575 872 5 2 93 0
0 0      0 937788 61568 366100 0 0 0 0 695 1551 2 2 96 0
0 0      0 937292 61568 366100 0 0 0 0 828 1863 10 4 87 0
1 0      0 937292 61568 366100 0 0 0 0 372 1292 1 1 98 0
0 0      0 937292 61568 366100 0 0 0 708 297 1222 1 1 99 0
0 0      0 937292 61576 366100 0 0 0 40 173 1204 0 1 82 18
1 0      0 937292 61576 366100 0 0 0 0 195 1239 1 1 98 0
1 0      0 937344 61576 366100 0 0 0 0 557 808 7 9 84 0
0 0      0 937368 61576 366100 0 0 0 0 232 1339 1 1 98 0
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
3 1      0 937368 61640 366036 0 0 0 164 410 1661 3 3 80 15
0 0      0 937368 61688 366056 0 0 0 104 399 1636 4 3 86 8
0 0      0 937368 61688 366100 0 0 0 0 182 1240 0 0 100 0
1 0      0 937368 61688 366100 0 0 0 0 213 1311 1 1 98 0
0 0      0 937368 61688 366100 0 0 0 0 184 1252 1 1 99 0
0 0      0 937368 61688 366100 0 0 0 0 193 1216 0 0 100 0
1 0      0 937368 61696 366096 0 0 0 76 214 1210 1 0 92 6
0 0      0 937336 61696 366100 0 0 0 0 179 1228 1 0 99 0
0 0      0 937336 61696 366100 0 0 0 0 201 1255 1 1 99 0
1 0      0 937336 61696 366100 0 0 0 0 179 1220 1 1 99 0
0 0      0 937336 61696 366100 0 0 0 0 165 1215 0 1 99 0
0 0      0 937336 61696 366100 0 0 0 36 167 1210 1 0 99 0
1 0      0 937336 61704 366100 0 0 0 16 194 1261 1 1 94 5
0 0      0 937336 61704 366100 0 0 0 0 258 1364 2 1 98 0
0 0      0 937336 61704 366100 0 0 0 0 198 1258 1 1 98 0
1 0      0 937416 61704 366100 0 0 0 0 211 1285 2 1 98 0
0 0      0 937432 61704 366100 0 0 0 0 191 1247 0 0 100 0
0 0      0 937432 61704 366100 0 0 0 4 244 1329 1 1 99 0
0 1      0 934992 61752 368860 0 0 2808 28 10131 13381 15 36 35 14
1 0      0 925480 61764 378196 0 0 9400 0 17849 28759 18 39 33 10
1 0      0 916444 61764 387012 0 0 8812 0 11275 10349 18 35 36 11
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
4 1      0 904536 61784 397544 0 0 10612 0 18607 32485 19 40 34 7
3 0      0 893888 61784 407448 0 0 9756 0 17270 33391 17 45 30 8
3 0      0 882836 61784 418156 0 0 10804 0 21894 43761 16 46 35 3
4 0      0 872608 61792 428564 0 0 10592 16 21012 41195 17 40 35 8
6 0      0 868648 61828 434220 0 0 5480 20 14549 20510 17 41 33 8
4 0      0 860812 61852 440248 0 0 6044 52 18770 32605 16 48 29 7
3 2      0 855184 61912 447332 0 0 7168 0 17802 34033 18 47 32 3
3 6      0 842888 61916 459716 0 0 12336 0 22642 45793 19 45 32 4
2 1      0 840436 61956 462640 0 0 2960 28 14272 18658 16 49 29 6

```

```

3 0      0 837784 62068 464548 0 0 1984 140 8841 9958 17 42 25 16
0 0      0 833824 62124 468608 0 0 4040 92 10523 13225 15 44 30 11
2 0      0 829388 62224 472632 0 0 4160 76 10541 10828 15 47 26 13
0 1      0 827164 62340 475372 0 0 2764 140 8321 8319 19 34 29 18
0 0      0 825856 62364 475732 0 0 252 56 1646 2108 7 10 73 10
0 0      0 825792 62388 475740 0 0 0 84 327 1506 3 2 89 7
0 0      0 825512 62388 475764 0 0 0 0 333 1611 9 2 90 0
2 0      0 819944 62388 475664 0 0 0 0 1151 1982 32 12 56 0
1 0      0 819084 62388 475608 0 0 0 0 1014 3325 47 9 44 0
0 0      0 819736 62388 475556 0 0 0 0 620 2146 10 4 86 0
0 0      0 819736 62396 475524 0 0 0 120 363 1588 5 3 86 7
0 0      0 819364 62396 475532 0 0 0 0 431 1798 8 2 90 0
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
2 0      0 819736 62396 475532 0 0 0 0 322 1524 3 1 96 0
2 0      0 819736 62396 475532 0 0 0 0 513 1985 6 3 91 0
0 0      0 824696 62404 475524 0 0 0 60 735 1781 33 9 54 5
0 0      0 824736 62444 475468 0 0 0 332 958 1028 10 3 75 12
1 0      0 824596 62444 475508 0 0 0 0 816 781 12 4 85 0
0 0      0 824480 62444 475508 0 0 0 0 510 1467 3 1 96 0
0 0      0 825720 62444 475512 0 0 0 0 318 1538 3 2 96 0
0 0      0 825720 62444 475512 0 0 0 0 266 1328 3 0 97 0
0 1      0 825720 62452 475504 0 0 0 216 368 1442 3 1 88 9
0 0      0 825720 62452 475512 0 0 0 0 706 1260 2 2 95 2
0 0      0 825720 62452 475512 0 0 0 0 783 2104 10 2 89 0
0 0      0 825232 62452 475512 0 0 0 0 619 5579 3 1 96 0
0 0      0 825232 62452 475512 0 0 0 0 193 2805 0 0 100 0
0 0      0 825232 62452 475512 0 0 0 0 202 1271 0 1 99 0
0 0      0 825232 62460 475512 0 0 0 60 198 1244 1 0 92 7
0 0      0 825232 62460 475512 0 0 0 0 185 1224 1 0 99 0
0 0      0 825728 62460 475512 0 0 0 0 334 1511 3 1 96 0
0 0      0 825728 62460 475512 0 0 0 0 238 1340 3 1 96 0
0 1      0 825728 62468 475504 0 0 0 24 259 1391 2 1 93 5
0 0      0 825696 62468 475512 0 0 0 0 218 1308 2 0 98 0
0 0      0 825696 62468 475512 0 0 0 0 193 1224 1 1 99 0
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1 0      0 825696 62468 475512 0 0 0 0 340 1558 2 2 97 0
1 0      0 825712 62468 475512 0 0 0 0 228 1359 2 1 97 0
0 1      0 825696 62476 475504 0 0 0 24 190 1238 0 1 94 5
0 0      0 825696 62476 475512 0 0 0 0 211 1287 2 1 97 1
1 0      0 825696 62476 475512 0 0 0 0 176 1215 0 0 100 0
0 0      0 825696 62476 475512 0 0 0 0 210 1256 1 0 99 0
0 0      0 825696 62476 475512 0 0 0 0 183 1201 0 1 99 0
1 0      0 825696 62476 475512 0 0 0 0 206 1304 1 1 98 0
0 0      0 825696 62484 475512 0 0 0 36 240 1339 1 1 91 7
0 0      0 825680 62484 475512 0 0 0 0 219 1283 1 1 98 0
0 0      0 825696 62484 475512 0 0 0 0 247 1384 1 1 98 0
0 0      0 825696 62484 475512 0 0 0 0 226 1311 1 1 98 0
0 0      0 825696 62484 475512 0 0 0 0 165 1195 0 0 100 0
0 0      0 825696 62484 475512 0 0 0 4 200 1266 1 0 99 0
0 0      0 825696 62492 475512 0 0 0 16 229 1306 1 1 94 5
4 0      0 816252 62492 483312 0 0 7908 0 12468 19614 15 31 43 11
1 0      0 804760 62492 494988 0 0 11520 0 25336 49376 18 45 36 1
0 0      0 792912 62492 506508 0 0 11520 0 25890 62564 15 52 33 0
2 0      0 786404 62532 512888 0 0 6572 32 16183 24631 16 43 35 6
3 0      0 786388 62688 514420 0 0 1648 120 7993 10956 15 31 42 13
0 1      0 784740 62740 516276 0 0 1900 52 3619 4449 12 18 54 15
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1 1      0 781532 62872 519180 0 0 2288 324 3849 4014 13 16 45 25
0 0      0 781448 62908 519172 0 0 64 44 787 1186 5 3 87 5
4 0      0 781056 62956 519396 0 0 0 480 1316 2661 7 3 79 12
0 0      0 781056 62956 519408 0 0 0 0 501 1244 6 3 91 0
1 0      0 775236 62956 519572 0 0 0 0 1398 3106 33 11 56 0
2 0      0 773644 62956 519544 0 0 0 0 891 2925 49 9 42 0
0 0      0 773816 62956 519492 0 0 0 0 667 1901 11 5 84 0
0 0      0 774824 62956 519468 0 0 0 68 330 1537 6 1 93 0
0 0      0 774824 62964 519468 0 0 0 32 378 626 5 3 88 5
0 0      0 774824 62964 519468 0 0 0 0 390 1722 6 2 92 0
0 0      0 774700 62964 519468 0 0 64 0 812 2068 5 4 87 4
2 0      0 780328 62972 519524 0 0 0 60 641 2610 31 7 57 5
1 1      0 779196 62996 519484 0 0 0 200 969 3375 18 4 71 7
0 0      0 779816 63012 519484 0 0 0 136 713 1585 4 2 90 5
0 0      0 779816 63012 519508 0 0 0 0 381 1496 3 1 96 0
1 0      0 779816 63012 519512 0 0 0 0 245 1360 1 1 98 0
0 0      0 779816 63012 519512 0 0 0 0 244 1384 1 1 98 0

```

0	0	0	779816	63012	519512	0	0	0	0	202	1313	1	1	99	0
1	0	0	779848	63020	519508	0	0	0	188	285	1421	1	1	88	11
0	0	0	779864	63020	519512	0	0	0	0	682	1595	2	2	97	0
0	0	0	779864	63020	519512	0	0	0	0	664	1442	4	1	95	0

ANEXO M - Teste de Processador – Comando vmstat - Servidor DRBL  
Etapas de 2 a 4 – Thin Client 01 e 02

```

procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1 0    0 1061988 54208 262792  0  0 1476  24  318 1151 18  6 62 13
0 0    0 1062004 54208 262820  0  0  0  0  216 299  1  1 99  0
1 0    0 1062144 54208 262820  0  0  0  0  258 359  1  1 99  0
0 0    0 1062168 54208 262820  0  0  0  0  203 284  1  1 98  0
0 0    0 1062136 54208 262820  0  0  0  0  204 248  1  0 99  0
0 1    0 1062136 54208 262820  0  0  0  20  241 327  1  0 99  1
1 0    0 1061920 54232 262800  0  0 100  72  695 1058  3  3 74 20
0 0    0 1061772 54240 262912  0  0  0  32  635 1944  2  4 93  2
2 0    0 1060600 54240 264328  0  0 1392  0 4699 7091  5 16 75  4
2 0    0 1058368 54240 266500  0  0 2176  0 8018 11437  9 32 60  0
1 0    0 1056136 54240 268576  0  0 2188  0 8100 11336  6 31 63  0
1 0    0 1054028 54240 270864  0  0 2176  0 8166 11619  7 29 63  0
2 0    0 1051912 54248 273032  0  0 2176  24 8211 11688 10 29 60  1
1 0    0 1049680 54248 275188  0  0 2176  0 8372 11512 15 30 55  0
4 0    0 1047696 54248 277164  0  0 2048  0 8312 11456  9 31 60  0
5 0    0 1045456 54248 279340  0  0 2176  0 7863 10760 17 34 50  0
4 0    0 1043220 54248 281388  0  0 2048  0 7655 10772 17 33 50  0
3 0    0 1042352 54248 282364  0  0  952  0 6587 9694 19 28 53  0
3 0    0 1042360 54256 282420  0  0  0  16 5040 7990 20 23 52  5
0 0    0 1042384 54256 282420  0  0  0  0 1392 2660  7  6 87  0
0 0    0 1042368 54256 282420  0  0  0  0  728 1586  7  2 92  0
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1 0    0 1042416 54256 282420  0  0  0  0  917 2020 11  3 86  0
0 0    0 1037484 54464 285320  0  0 3144  32 3025 2566  9 10 62 18
2 1    0 1036140 55020 286964  0  0 2220  0 4004 4732 15 19 47 19
0 1    0 1034760 55296 288028  0  0 1228 172 2972 4099 14 17 51 19
0 1    0 1033124 55468 289636  0  0 1768  56 4481 5605 14 24 48 14
0 2    0 1030068 55680 292172  0  0 2780 140 6988 7543 16 31 18 36
3 1    0 1028444 55896 292644  0  0  428  628 2991 4187 12 17 40 31
4 9    0 1027904 55992 294432  0  0 1868  224 9534 11234 14 41 23 21
0 12   0 1023140 56304 298644  0  0 4292 152 9473 10154 18 38  8 36
0 1    0 1019380 56464 301952  0  0 3488 180 10032 14529 15 35  7 43
1 2    0 1017368 56604 303824  0  0 2004 300 8012 7554 18 31 27 25
2 1    0 1015456 56836 305048  0  0 1284 532 5313 6094 14 25 22 39
5 0    0 1014852 56940 305652  0  0  268 560 5389 6194 15 26 45 15
1 2    0 1011900 57024 307056  0  0 1360 296 8177 8014 14 42 26 18
2 5    0 1011072 57312 308480  0  0 1668 340 9469 9643 19 43  9 29
0 3    0 1004240 57456 311292  0  0 2792 364 7639 6672 25 33  5 37
1 0    0 1003868 57616 315040  0  0 3788 288 9950 15995 21 34 10 36
2 1    0 1000364 57904 315736  0  0  864 756 9793 10028 13 48  9 30
2 0    0 1000620 58036 317300  0  0 1552 424 11246 12529 17 45 19 19
2 0    0 998976 58056 318868  0  0 1488  24 11328 13760 17 41 39  4
3 0    0 997548 58208 319844  0  0 1116 296 8780 10739 14 38 27 21
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
3 0    0 997172 58376 319940  0  0 152 356 8714 8902 16 41 24 19
1 1    0 996872 58580 320308  0  0 252 592 7992 8553 16 34 28 22
2 1    0 996420 58692 321016  0  0  808 236 10963 16829 16 41 28 14
1 0    0 994544 58864 321960  0  0 772 612 10030 8508 14 45 19 22
2 2    0 993836 59060 322168  0  0 292 652 5972 6791 17 30 25 28
0 2    0 992944 59296 322620  0  0 184 896 4629 5615 16 25 22 37
3 4    0 987152 59368 328988  0  0 6300 204 20316 37058 13 52 21 14
1 6    0 984440 59528 331244  0  0 2468 336 12330 22873 17 43 16 24
0 5    0 983776 59740 332052  0  0 796 548 7983 7336 17 37 11 35
6 2    0 981788 59960 333620  0  0 1528 572 7409 7332 18 38 14 29
3 1    0 978952 60200 333796  0  0 232 852 11428 18309 18 39 17 25
3 4    0 977212 60340 335816  0  0 2100 484 22412 45985 15 53 22 10
3 3    0 977552 60596 337004  0  0 1120 468 11077 9151 17 46 14 23
1 0    0 977148 60756 337112  0  0 308 536 9980 12617 17 43 17 23
2 1    0 974808 60920 337892  0  0 696 400 10166 12118 18 39 27 17
0 0    0 975212 60984 337976  0  0  92 128 6600 7124 16 31 45  9
3 0    0 974012 61040 338044  0  0  72 32 6404 8823 14 30 51  5
3 0    0 975076 61152 338012  0  0  44 200 3794 5954 14 23 44 20
2 0    0 973712 61428 338064  0  0 296  68 11158 17354 16 38 41  5
0 1    0 974364 61536 338160  0  0  24 240 5424 5691 10 23 52 15
1 2    0 973696 61632 338084  0  0  44 276 6114 7263 15 34 41 10
procs -----memory----- --swap-- ----io---- -system-- ----cpu----
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa

```

```

2 2      0 973448 61872 338240    0    0    16    724 4354 6110 13 24 33 29
1 0      0 971784 61980 338184    0    0     4    324 14347 29781 16 45 27 13
3 1      0 973420 62168 338248    0    0    92    520 10350 20455 15 37 24 25
3 1      0 972724 62372 338644    0    0   452    780 5246 6259 17 26 21 36
2 1      0 972156 62532 338684    0    0   180    704 6011 6404 16 31 30 23
1 1      0 971908 62792 338904    0    0     8    832 3020 3732 14 18 38 30
2 1      0 971448 62868 338836    0    0    52    260 9113 7749 17 38 37  8
0 3      0 971272 62984 339472    0    0   616    140 6723 6664 12 32 47  8
2 0      0 971200 63056 339628    0    0   156    112 4230 4897 14 20 56 11
1 0      0 970160 63172 339624    0    0    92    176 4750 5288 17 24 41 17
0 0      0 970068 63236 339760    0    0     4    100 2251 4282 13 14 59 14
0 0      0 970100 63256 339760    0    0     8     52  313 1491  3  1 87  9
0 0      0 970100 63256 339800    0    0     0     0  294 1321  3  1 96  0
0 0      0 970100 63256 339800    0    0     0     0 1269 2980  6  7 87  0
0 0      0 970068 63312 339768    0    0     0   188 1122 1792  7  7 72 15
0 0      0 970100 63320 339824    0    0     0     64  279 1353  1  1 95  3
0 0      0 970100 63360 339816    0    0     0   128 1045 2541  7  6 76 12
0 0      0 970116 63360 339820    0    0     0     0  172 1217  0  0 100  0
0 0      0 970100 63360 339824    0    0     0     0  315 1395  4  1 95  0
0 0      0 970068 63360 339824    0    0     0     0  289 1516  3  1 96  0
0 0      0 970052 63468 339740    0    0     4    228  609 2022  5  5 75 15
procs -----memory----- ---swap--- -----io----- -system-- -----cpu-----
r  b      swpd   free   buff  cache    si   so    bi    bo    in    cs us sy id wa
1  1      0 969712 63572 339724    0    0     0   472 2336 3937 12 15 61 13
5  1      0 969324 63784 339780    0    0    68    720 3504 5879 14 22 40 24
1  0      0 969076 63900 339916    0    0   196    264 3773 6306 15 24 42 19
2  4      0 968116 64020 340076    0    0   248    192 6089 7425 17 34 36 13
4  0      0 967688 64152 340200    0    0   168    312 6119 7489 15 35 36 15
3  0      0 966288 64240 340888    0    0   720    172 8181 8292 17 41 35  8
3  0      0 965412 64512 341364    0    0   404    464 5776 7210 14 32 25 29
0  0      0 962764 64572 343960    0    0  2720    152 13852 18479 17 44 31  8
1  1      0 960812 64664 347056    0    0  3132    236 13241 16987 14 43 28 14
5  1      0 954188 64740 352060    0    0  4952    116 15328 23442 17 44 25 14
1  0      0 953316 64760 353360    0    0  1296     0 12740 22287 17 39 42  2
0  0      0 953108 64784 354892    0    0  1568    32 7328 10714 15 28 47  9
0  0      0 952416 64856 354896    0    0    68    152 5463 5904 12 27 54  6
1  1      0 952148 65136 355020    0    0   120    624 6010 7562 17 32 31 21
4  0      0 950000 65260 355176    0    0   284    216 5095 5882 11 29 47 14
3  1      0 951084 65368 355364    0    0    80    224 12917 19313 16 46 32  7
4  1      0 950576 65460 356852    0    0  1584    152 12952 20470 18 43 29 10
4  2      0 949328 65580 357104    0    0   292    184 15430 26512 16 51 26  8
4  1      0 949236 65624 357356    0    0   168    100 9770 8997 17 39 33 11
1  0      0 949304 65656 357388    0    0     0     80 14004 23496 16 39 41  4
0  0      0 948752 65656 357388    0    0     0     0  7616 11730 16 29 55  0
procs -----memory----- ---swap--- -----io----- -system-- -----cpu-----
r  b      swpd   free   buff  cache    si   so    bi    bo    in    cs us sy id wa
3  0      0 949204 65732 357344    0    0    16    152 5391 6990 12 19 61  8
3  0      0 945136 65784 360544    0    0  3288    68 10146 19866 12 34 46  8
2  4      0 937432 65856 368996    0    0  8384    212 17947 31232 15 44 30 11
0  0      0 936016 65884 371064    0    0  2052    80 7215 14587  7 18 69  7
0  0      0 936096 65892 371084    0    0     0     48 1345 1855  7  6 84  4
0  0      0 936072 65896 371088    0    0     4     0 1652 2621  9  9 82  0
1  0      0 934416 65944 372548    0    0  1544    128 8452 9946 19 32 39 10
1  1      0 934056 65968 372756    0    0   116    16 3640 6121 13 25 56  7
0  1      0 934072 65988 372752    0    0    20    28 3844 5756 13 10 68  9
3  0      0 932432 66012 372752    0    0    24    24 8426 14066 14 28 54  4
2  1      0 932324 66060 372752    0    0     0   148 12814 23660 18 33 39 10
0  0      0 934272 66072 372772    0    0     4    20 11766 20205 18 29 49  4
1  0      0 933344 66072 372784    0    0     0     0 8840 13583 16 29 55  0
3  1      0 933084 66132 372736    0    0    40    64 5254 6095 14 30 45 12
2  0      0 932260 66160 372784    0    0    12    52 3454 4316 21 21 53  4
3  0      0 932484 66208 373756    0    0   976    132 4708 5911 17 22 45 17
5  1      0 931684 66236 373776    0    0    12    44 6076 8199 15 24 53  8
2  0      0 922696 66244 382096    0    0  8336     0 15492 30371 17 41 36  6
0  0      0 920372 66324 385940    0    0  3860    344 12670 20242 13 34 39 13
6  0      0 920372 66332 386032    0    0     0    32  316  551  4  1 90  5
1  0      0 912464 66368 386672    0    0   176     0 1379 3215 45 10 39  5
procs -----memory----- ---swap--- -----io----- -system-- -----cpu-----
r  b      swpd   free   buff  cache    si   so    bi    bo    in    cs us sy id wa
5  0      0 911664 66380 386668    0    0    44     0 1018 4443 40  9 51  0
0  0      0 912076 66388 386636    0    0     0    52  592 1115 11  4 84  2
0  0      0 912076 66396 386636    0    0     0    20  480 1882  9  2 85  5
0  0      0 912108 66396 386644    0    0     0     0  230 1331  2  1 98  0
0  0      0 912108 66396 386644    0    0     0     0  229 1328  2  1 97  0
4  0      0 916696 66404 386636    0    0     0    60 1105 2172 32 14 50  4
3  1      0 917612 66456 386568    0    0     0    204 1056 2458 15  6 66 13
0  1      0 916496 66568 386620    0    0    24    472 1141 2332 11  4 64 20

```

```

0 0      0 916744 66616 386688 0 0 12 152 861 958 7 4 77 13
0 0      0 916744 66616 386712 0 0 0 0 323 537 3 1 96 0
0 0      0 916744 66616 386712 0 0 0 0 226 330 2 1 97 0
0 0      0 916744 66616 386712 0 0 0 0 236 1104 2 0 98 0
0 0      0 916744 66616 386712 0 0 0 0 758 646 5 2 93 0
0 0      0 916744 66620 386708 0 0 4 1000 971 3285 11 2 83 4
0 0      0 916372 66628 386720 0 0 0 60 574 901 2 1 75 23
0 0      0 916372 66628 386720 0 0 0 0 211 243 1 1 98 0
0 0      0 916340 66628 386720 0 0 0 0 192 222 1 0 99 0
0 0      0 916340 66628 386720 0 0 0 0 212 1277 1 1 99 0
0 0      0 916992 66628 386724 0 0 0 0 226 1278 1 1 99 0
0 0      0 916868 66628 386724 0 0 0 44 248 314 2 1 98 0
0 0      0 916992 66636 386724 0 0 0 24 364 530 4 1 91 5
procs -----memory----- ---swap--- ---io--- ---system--- ---cpu---
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
3 1  0 916992 66644 386716 0 0 0 24 227 1316 2 2 93 4
0 0  0 916976 66644 386720 0 0 0 0 259 385 3 1 95 1
0 0  0 916992 66644 386724 0 0 0 0 215 263 2 1 98 0
1 0  0 916992 66644 386724 0 0 0 0 276 1443 3 1 97 0
2 3  0 913948 66712 389332 0 0 2820 60 10971 14495 15 37 33 15
4 3  0 905144 66808 398008 0 0 8632 204 18055 25058 16 39 23 23
3 2  0 896328 66820 406420 0 0 8404 0 11302 11095 15 41 32 11
3 5  0 885392 66828 417596 0 0 11068 0 17368 30155 16 40 36 8
3 0  0 878092 66828 424956 0 0 7468 0 13413 21016 16 41 32 11
4 0  0 864496 66836 436756 0 0 11700 52 21930 44309 13 48 32 7
1 7  0 854668 66836 448212 0 0 11468 0 21192 38442 17 43 34 5
3 2  0 848004 66904 454672 0 0 6540 176 18406 28352 17 46 26 12
2 0  0 846692 66964 455412 0 0 676 120 12848 16932 15 48 29 8
3 0  0 844748 66972 455800 0 0 468 20 19477 42560 16 49 30 5
0 0  0 846148 66980 455992 0 0 108 20 21198 44914 21 43 34 3
3 0  0 838632 67052 463136 0 0 7260 0 18458 34612 17 49 27 7
1 2  0 833624 67080 469188 0 0 6068 0 17485 31787 19 45 32 5
2 0  0 830352 67100 471308 0 0 2144 0 15386 29514 16 52 30 3
3 0  0 829924 67140 471596 0 0 308 40 14901 25619 15 51 30 4
1 0  0 828888 67192 472400 0 0 852 144 20100 40051 18 49 27 6
4 0  0 827896 67224 472932 0 0 528 68 19921 42826 17 49 30 4
procs -----memory----- ---swap--- ---io--- ---system--- ---cpu---
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
5 1  0 829344 67244 473264 0 0 328 0 18015 41278 12 54 33 1
2 1  0 825004 67296 477556 0 0 4300 56 13196 13930 15 50 27 8
1 0  0 820316 67432 481416 0 0 3940 40 11618 12386 17 46 25 11
2 1  0 815660 67552 483368 0 0 2068 132 15301 22218 18 47 25 10
0 0  0 816756 67572 484440 0 0 924 24 19041 40416 16 52 29 4
2 0  0 816636 67600 484412 0 0 4 92 8568 12423 19 32 38 11
1 1  0 816628 67660 484380 0 0 52 76 6074 7395 13 34 44 8
1 1  0 816580 67712 484404 0 0 8 132 6419 7271 15 33 42 9
0 0  0 816516 67756 484428 0 0 12 116 5387 6130 15 28 49 7
2 0  0 816408 67760 484452 0 0 40 0 4072 5266 15 23 59 3
0 0  0 816192 67760 484492 0 0 0 0 6651 8691 16 24 60 0
1 0  0 816344 67792 484468 0 0 24 40 5188 5885 16 25 53 6
3 1  0 815724 67840 485452 0 0 976 76 5117 6008 14 29 46 12
5 0  0 814952 67868 485620 0 0 152 56 4553 5324 15 27 55 2
2 0  0 815068 67896 485680 0 0 12 84 3770 5111 14 22 58 6
0 0  0 814796 67904 485708 0 0 0 52 3664 4853 15 17 65 3
1 0  0 815068 67904 485716 0 0 0 0 4516 6283 16 23 61 0
2 0  0 814748 67920 485700 0 0 0 40 4407 6316 14 27 55 4
0 0  0 814768 67924 485712 0 0 4 0 3114 4367 12 16 68 4
0 0  0 814412 67924 485716 0 0 0 0 390 1686 10 3 88 0
1 0  0 809248 67924 485516 0 0 0 0 1258 2227 35 10 55 0
procs -----memory----- ---swap--- ---io--- ---system--- ---cpu---
r b  swpd  free  buff  cache  si  so  bi  bo  in  cs  us  sy  id  wa
1 0  0 808628 67924 485588 0 0 0 0 879 3197 47 8 45 0
0 0  0 808660 67924 485484 0 0 0 0 604 2237 11 4 85 0
2 0  0 809280 67932 485484 0 0 0 32 407 659 6 1 88 4
0 0  0 809280 67932 485484 0 0 0 0 382 1645 7 1 92 0
1 0  0 809280 67932 485484 0 0 0 0 241 1370 2 1 97 0
1 0  0 808784 67940 485476 0 0 0 32 528 1936 21 3 72 4
0 0  0 813784 67948 485464 0 0 0 56 925 1777 23 5 68 4
1 0  0 814404 67988 485440 0 0 0 332 883 1924 6 2 79 13
0 0  0 814032 67988 485464 0 0 0 0 719 582 8 3 90 0
0 0  0 814776 67988 485468 0 0 0 0 395 1498 2 1 97 0
1 0  0 814776 67988 485468 0 0 0 0 349 552 4 1 95 0
0 0  0 814776 67988 485468 0 0 0 0 257 1387 2 1 97 0
0 0  0 814776 67996 485460 0 0 0 328 743 1734 4 2 83 11
0 0  0 814776 67996 485468 0 0 0 0 819 2593 11 3 86 0
0 0  0 814776 67996 485468 0 0 0 0 459 1266 2 0 98 0
0 0  0 814776 68004 485460 0 0 0 36 264 1423 2 2 92 5

```

```

1 0      0 814776 68004 485468      0 0      0 0 285 1448 1 0 99 0
2 0      0 814776 68004 485468      0 0      0 0 382 1603 4 2 94 0
0 0      0 814776 68012 485460      0 0      0 24 275 421 3 1 92 5
0 0      0 814776 68012 485468      0 0      0 0 199 1272 1 1 98 0
0 0      0 814404 68012 485468      0 0      0 0 289 405 2 1 97 0
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free  buff  cache      si  so  bi  bo  in  cs  us  sy  id  wa
1 1      0 814776 68012 485468      0 0      4 0 368 1601 3 2 92 3
3 0      0 803536 68012 495172      0 0 9696 0 15775 27042 16 39 38 7
5 0      0 791356 68020 506944      0 0 11760 24 25776 51269 15 50 33 2
0 0      0 781072 68020 517800      0 0 10868 0 23336 49605 17 49 34 1
2 0      0 775888 68060 523596      0 0 5964 32 15027 27866 16 42 36 6
3 0      0 774916 68224 524952      0 0 1424 120 5110 6209 16 26 41 17
3 0      0 773584 68284 526236      0 0 1368 52 3397 3899 14 16 56 13
0 0      0 770588 68408 529020      0 0 2304 340 3921 3981 12 17 43 27
2 0      0 769668 68436 529196      0 0 32 20 2680 4135 7 13 77 3
3 1      0 768532 68468 529264      0 0 0 432 20678 42941 17 45 34 4
3 0      0 768288 68484 529324      0 0 0 60 21764 44287 16 49 33 2
3 0      0 770368 68496 529312      0 0 16 0 15181 33510 22 37 38 3
2 1      0 770272 68520 529312      0 0 16 32 12011 23853 14 33 48 5
0 0      0 769804 68540 529956      0 0 656 40 4909 6174 17 24 50 9
2 0      0 769548 68552 530080      0 0 80 24 2745 3980 13 12 71 3
0 0      0 769456 68552 530056      0 0 44 0 895 2504 8 4 79 9
0 0      0 768096 68632 531472      0 0 952 232 2964 3411 13 16 59 12
0 0      0 767396 68684 531960      0 0 452 128 1605 2355 9 9 73 9
1 0      0 766968 68724 532192      0 0 0 468 1377 2844 4 6 80 10
0 0      0 766904 68724 532160      0 0 0 0 439 1599 2 3 96 0
0 0      0 766920 68724 532160      0 0 0 0 270 1453 8 1 91 0
procs -----memory----- ---swap--- ----io---- -system-- ----cpu----
r b      swpd  free  buff  cache      si  so  bi  bo  in  cs  us  sy  id  wa
5 0      0 762540 68724 532160      0 0 0 0 759 2567 20 7 73 0
2 0      0 759328 68724 532360      0 0 0 0 1237 3269 57 10 32 0
0 0      0 759624 68724 532280      0 0 0 120 862 2565 14 7 79 0
0 0      0 760516 68732 532256      0 0 0 24 386 1654 7 1 85 8
0 0      0 760516 68732 532256      0 0 0 0 359 1562 6 2 92 0
2 1      0 760516 68740 532248      0 0 32 32 275 1478 3 1 88 9
2 0      0 760492 68740 532324      0 0 32 0 779 1991 4 4 93 0
1 0      0 765384 68748 532216      0 0 0 72 769 1923 36 6 53 5
0 1      0 765500 68788 532260      0 0 0 320 818 6100 7 6 78 9
1 0      0 766444 68788 532300      0 0 0 8 764 1848 6 2 90 2
0 0      0 766492 68788 532300      0 0 0 0 550 1482 3 1 96 0
0 0      0 766492 68788 532304      0 0 0 0 527 1539 2 1 97 0
1 0      0 766492 68788 532304      0 0 0 0 829 1864 6 3 91 0
0 0      0 766244 68788 532304      0 0 0 0 750 652 7 4 89 0

```

ANEXO N - Teste de tráfego de rede – Comando iptraf – Servidor LTSP  
Etapas de 2 a 4 – Thin Client 01

```

Terminal
Terminal
IPTraf
Statistics for eth0
-----
                Total      Total      Incoming  Incoming  Outgoing  Outgoing
                Packets    Bytes     Packets   Bytes     Packets   Bytes
Total:          94899    77827082   39229    3666014   55670    74161068
IP:             94899    76292793   39229    2911105   55670    73381688
TCP:            64730    54574589   23790    2345773   40940    52228816
UDP:            3015     21715880   15439    565332    14712    21150548
ICMP:           18         2324       0         0         18         2324
Other IP:       0         0         0         0         0         0
Non-IP:         0         0         0         0         0         0

Total rates:          9,7 kbits/sec      Broadcast packets:      4
                   15,0 packets/sec      Broadcast bytes:      1818

Incoming rates:       4,2 kbits/sec
                   5,0 packets/sec

Outgoing rates:       5,4 kbits/sec
                   10,0 packets/sec

Elapsed time: 0:02
X-exit

Terminal
IPTraf
Statistics for eth0
-----
                Total      Total      Incoming  Incoming  Outgoing  Outgoing
                Packets    Bytes     Packets   Bytes     Packets   Bytes
Total:         105866    81971352   44134    4257130   61732   77714222
IP:            105866    80283447   44134    3433473   61732   76849974
TCP:            75644    58558999   28642    2861982   47002   55697017
UDP:            30204    21722039   15492    571491    14712   21150548
ICMP:           18         2409       0         0         18         2409
Other IP:       0         0         0         0         0         0
Non-IP:         0         0         0         0         0         0

Total rates:        1689,2 kbits/sec      Broadcast packets:      4
                   455,2 packets/sec      Broadcast bytes:      1818

Incoming rates:     173,4 kbits/sec
                   184,8 packets/sec

Outgoing rates:     1515,8 kbits/sec
                   270,4 packets/sec

Elapsed time: 0:02
X-exit

Terminal
IPTraf
Statistics for eth0
-----
                Total      Total      Incoming  Incoming  Outgoing  Outgoing
                Packets    Bytes     Packets   Bytes     Packets   Bytes
Total:         136727    99247113   56699    5468082   80028   93779031
IP:            136727    97127232   56699    4468593   80028   92658639
TCP:            106557    75408961   41260    3903261   65297   71505700
UDP:            30152    21715947   15439    565332    14713   21150615
ICMP:           18         2324       0         0         18         2324
Other IP:       0         0         0         0         0         0
Non-IP:         0         0         0         0         0         0

Total rates:         11,6 kbits/sec      Broadcast packets:      4
                   18,0 packets/sec      Broadcast bytes:      1818

Incoming rates:       5,0 kbits/sec
                   6,0 packets/sec

Outgoing rates:       6,6 kbits/sec
                   12,0 packets/sec

Elapsed time: 0:00
X-exit

```

ANEXO O - Teste de tráfego de rede – Comando iptraf – Servidor LTSP  
Etapas de 2 a 4 – Thin Client 02

The image displays three sequential screenshots of a terminal window running the IPtraf command to monitor network traffic on the eth0 interface. Each screenshot shows a table of statistics and summary rates.

**Terminal 1 (Elapsed time: 0:00)**

	Total Packets	Total Bytes	Incoming Packets	Incoming Bytes	Outgoing Packets	Outgoing Bytes
<b>Total:</b>	91014	73475746	38347	3580918	52667	69894828
<b>IP:</b>	91014	71995847	38347	2838357	52667	69157490
<b>TCP:</b>	60706	50260795	22771	2256383	37935	48004412
<b>UDP:</b>	30288	21732486	15576	581974	14712	21150512
<b>ICMP:</b>	20	2566	0	0	20	2566
<b>Other IP:</b>	0	0	0	0	0	0
<b>Non-IP:</b>	0	0	0	0	0	0

Summary rates:  
 Total rates: 51,2 kbits/sec, 38,4 packets/sec  
 Incoming rates: 34,4 kbits/sec, 18,6 packets/sec  
 Outgoing rates: 16,8 kbits/sec, 19,8 packets/sec  
 Broadcast packets: 4, Broadcast bytes: 1818  
 IP checksum errors: 0

**Terminal 2 (Elapsed time: 0:01)**

	Total Packets	Total Bytes	Incoming Packets	Incoming Bytes	Outgoing Packets	Outgoing Bytes
<b>Total:</b>	112739	83820876	47609	4560278	65130	79260598
<b>IP:</b>	112739	82036827	47609	3688049	65130	78348778
<b>TCP:</b>	82431	60301775	32033	3106075	50398	57195700
<b>UDP:</b>	30288	21732486	15576	581974	14712	21150512
<b>ICMP:</b>	20	2566	0	0	20	2566
<b>Other IP:</b>	0	0	0	0	0	0
<b>Non-IP:</b>	0	0	0	0	0	0

Summary rates:  
 Total rates: 23,7 kbits/sec, 19,2 packets/sec  
 Incoming rates: 6,0 kbits/sec, 8,2 packets/sec  
 Outgoing rates: 17,7 kbits/sec, 11,0 packets/sec  
 Broadcast packets: 4, Broadcast bytes: 1818  
 IP checksum errors: 0

**Terminal 3 (Elapsed time: 0:00)**

	Total Packets	Total Bytes	Incoming Packets	Incoming Bytes	Outgoing Packets	Outgoing Bytes
<b>Total:</b>	130063	91723616	55130	5378824	74933	86344792
<b>IP:</b>	130063	89697031	55130	4401301	74933	85295730
<b>TCP:</b>	99755	67961979	39554	3819327	60201	64142652
<b>UDP:</b>	30288	21732486	15576	581974	14712	21150512
<b>ICMP:</b>	20	2566	0	0	20	2566
<b>Other IP:</b>	0	0	0	0	0	0
<b>Non-IP:</b>	0	0	0	0	0	0

Summary rates:  
 Total rates: 1,9 kbits/sec, 3,0 packets/sec  
 Incoming rates: 0,8 kbits/sec, 1,0 packets/sec  
 Outgoing rates: 1,1 kbits/sec, 2,0 packets/sec  
 Broadcast packets: 4, Broadcast bytes: 1818  
 IP checksum errors: 0

ANEXO P - Teste de tráfego de rede – Comando iptraf – Servidor LTSP  
Etapas de 2 a 4 – Thin Client 01 e 02

The image displays three sequential terminal windows showing the output of the IPtraf command on interface eth0. Each window provides a snapshot of network traffic statistics, including total and incoming/outgoing packets and bytes, as well as rates in kbits/sec and packets/sec. The statistics are presented in a structured table format.

**Terminal 1 (Elapsed time: 0:01)**

	Total Packets	Total Bytes	Incoming Packets	Incoming Bytes	Outgoing Packets	Outgoing Bytes
Total:	181850	148447K	75730	7093545	106120	141354K
IP:	181850	145490K	75730	5621919	106120	139868K
TCP:	121335	102033K	44679	4471882	76656	97561815
UDP:	60474	43451030	31051	1150037	29423	42300993
ICMP:	41	5515	0	0	41	5515
Other IP:	0	0	0	0	0	0
Non-IP:	0	0	0	0	0	0

**Terminal 2 (Elapsed time: 0:00)**

	Total Packets	Total Bytes	Incoming Packets	Incoming Bytes	Outgoing Packets	Outgoing Bytes
Total:	254721	178766K	106847	10709415	147874	168057K
IP:	254721	174788K	106847	8802151	147874	165986K
TCP:	194189	131329K	75788	7650682	118401	123678K
UDP:	60483	43452529	31059	1151469	29424	42301060
ICMP:	49	7171	0	0	49	7171
Other IP:	0	0	0	0	0	0
Non-IP:	0	0	0	0	0	0

**Terminal 3 (Elapsed time: 0:10)**

	Total Packets	Total Bytes	Incoming Packets	Incoming Bytes	Outgoing Packets	Outgoing Bytes
Total:	374172	253805K	155419	17827260	218753	235978K
IP:	374172	248156K	155419	15239988	218753	232916K
TCP:	313634	204695K	124357	14087982	189277	190607K
UDP:	60486	43453066	31062	1152006	29424	42301060
ICMP:	52	7792	0	0	52	7792
Other IP:	0	0	0	0	0	0
Non-IP:	0	0	0	0	0	0

ANEXO Q - Teste de tráfego de rede – Comando iptraf – ServidorDRBL  
Etapas de 2 a 4 – Thin Client 01

```

paulo@drbl: ~
paulo@drbl: ~
paulo@drbl: ~
IPTraff
Statistics for eth0
-----
              Total      Total      Incoming  Incoming  Outgoing  Outgoing
              Packets    Bytes    Packets   Bytes    Packets   Bytes
Total:        498100    396547K    198563    21432533  299537    375115K
IP:           498100    389374K    198563    18453005  299537    370921K
TCP:          468886    368321K    183938    17954904  284948    350366K
UDP:           29214    21053561    14625     498101    14589     20555460
ICMP:           0         0         0         0         0         0
Other IP:      0         0         0         0         0         0
Non-IP:        0         0         0         0         0         0

Total rates:   6684,8 kbits/sec      Broadcast packets:    11
                1773,4 packets/sec      Broadcast bytes:      2253

Incoming rates: 1014,6 kbits/sec
                  806,6 packets/sec

Outgoing rates: 5670,2 kbits/sec
                  966,8 packets/sec

Elapsed time:  0:01
X-exit

paulo@drbl: ~
paulo@drbl: ~
paulo@drbl: ~
IPTraff
Statistics for eth0
-----
              Total      Total      Incoming  Incoming  Outgoing  Outgoing
              Packets    Bytes    Packets   Bytes    Packets   Bytes
Total:        552099    446753K    218137    23502660  333962    423251K
IP:           552099    438824K    218137    20249096  333962    418575K
TCP:          522858    417768K    203498    19749488  319360    398019K
UDP:           29241    21055968    14639     499608    14602     20556360
ICMP:           0         0         0         0         0         0
Other IP:      0         0         0         0         0         0
Non-IP:        0         0         0         0         0         0

Total rates:   6505,3 kbits/sec      Broadcast packets:    11
                974,0 packets/sec      Broadcast bytes:      2253

Incoming rates: 893,0 kbits/sec
                  404,6 packets/sec

Outgoing rates: 5612,3 kbits/sec
                  569,4 packets/sec

Elapsed time:  0:01
X-exit

paulo@drbl: ~
paulo@drbl: ~
paulo@drbl: ~
IPTraff
Statistics for eth0
-----
              Total      Total      Incoming  Incoming  Outgoing  Outgoing
              Packets    Bytes    Packets   Bytes    Packets   Bytes
Total:        337152    263121K    136746    14852544  200406    248269K
IP:           337152    258202K    136746    12738454  200406    245463K
TCP:          308265    237182K    122285    12258276  185980    224924K
UDP:           28887    21019578    14461     480178    14426     20539400
ICMP:           0         0         0         0         0         0
Other IP:      0         0         0         0         0         0
Non-IP:        0         0         0         0         0         0

Total rates:   46556,5 kbits/sec      Broadcast packets:    11
                6803,8 packets/sec      Broadcast bytes:      2253

Incoming rates: 2175,6 kbits/sec
                  2577,6 packets/sec

Outgoing rates: 44380,9 kbits/sec
                  4226,2 packets/sec

Elapsed time:  0:01
X-exit

```

ANEXO R - Teste de tráfego de rede – Comando iptraf – ServidorDRBL  
Etapas de 2 a 4 – Thin Client 02

```

paulo@drbl: ~
paulo@drbl: ~
paulo@drbl: ~
IPTraf
Statistics for eth0
-----
              Total      Total      Incoming  Incoming  Outgoing  Outgoing
              Packets    Bytes    Packets    Bytes    Packets    Bytes
Total:         313655    239483K    127036    13998523    186619    225484K
IP:            313655    234892K    127036    12020379    186619    222872K
TCP:          284806    213874K    112597    11542196    172209    202332K
UDP:           28849    21017690    14439    478183    14410    20539507
ICMP:           0         0         0         0         0         0
Other IP:      0         0         0         0         0         0
Non-IP:        0         0         0         0         0         0

Total rates:    6128,1 kbits/sec      Broadcast packets:    11
                1328,8 packets/sec      Broadcast bytes:     2253

Incoming rates:  721,9 kbits/sec
                 575,2 packets/sec

Outgoing rates:  5406,2 kbits/sec      IP checksum errors:    0
                 753,6 packets/sec

Elapsed time:  0:01
X-exit

paulo@drbl: ~
paulo@drbl: ~
paulo@drbl: ~
IPTraf
Statistics for eth0
-----
              Total      Total      Incoming  Incoming  Outgoing  Outgoing
              Packets    Bytes    Packets    Bytes    Packets    Bytes
Total:         498194    395143K    196758    21097078    301436    374046K
IP:            498194    387968K    196758    18142826    301436    369825K
TCP:          469008    366913K    182150    17646088    286858    349267K
UDP:           29186    21055193    14608    496738    14578    20558455
ICMP:           0         0         0         0         0         0
Other IP:      0         0         0         0         0         0
Non-IP:        0         0         0         0         0         0

Total rates:    45815,2 kbits/sec      Broadcast packets:    11
                9375,6 packets/sec      Broadcast bytes:     2253

Incoming rates:  4277,1 kbits/sec
                 4051,4 packets/sec

Outgoing rates:  41538,0 kbits/sec      IP checksum errors:    0
                 5324,2 packets/sec

Elapsed time:  0:02
X-exit

paulo@drbl: ~
paulo@drbl: ~
paulo@drbl: ~
IPTraf
Statistics for eth0
-----
              Total      Total      Incoming  Incoming  Outgoing  Outgoing
              Packets    Bytes    Packets    Bytes    Packets    Bytes
Total:         552091    445224K    216074    23159125    336017    422064K
IP:            552091    437295K    216074    19934449    336017    417360K
TCP:          522877    416237K    201452    19436192    321425    396801K
UDP:           29214    21057727    14622    498257    14592    20559470
ICMP:           0         0         0         0         0         0
Other IP:      0         0         0         0         0         0
Non-IP:        0         0         0         0         0         0

Total rates:    66490,6 kbits/sec      Broadcast packets:    11
                8779,2 packets/sec      Broadcast bytes:     2253

Incoming rates:  2081,2 kbits/sec
                 3091,2 packets/sec

Outgoing rates:  64420,0 kbits/sec      IP checksum errors:    0
                 5688,0 packets/sec

Elapsed time:  0:03
X-exit

```

ANEXO S - Teste de tráfego de rede – Comando iptraf – ServidorDRBL  
Etapas de 2 a 4 – Thin Client 01 e 02

```

paulo@drbl: ~
paulo@drbl: ~
paulo@drbl: ~
IPtraf
Statistics for eth0
-----
                Total      Total      Incoming  Incoming  Outgoing  Outgoing
                Packets    Bytes     Packets   Bytes     Packets   Bytes
Total:          676920    526593K   273656    29795080  403264    496798K
IP:             676920    516717K   273656    25564616  403264    491152K
TCP:           619169    474678K   244747    24605340  374422    450073K
UDP:           57750     42038745  28909     959276   28841     41079469
ICMP:           1         48        0         0         1         48
Other IP:       0         0         0         0         0         0
Non-IP:         0         0         0         0         0         0

Total rates:    33375,4 kbits/sec      Broadcast packets: 22
                4964,8 packets/sec      Broadcast bytes:   4506

Incoming rates: 1829,5 kbits/sec
                1886,6 packets/sec

Outgoing rates: 31545,9 kbits/sec
                3078,2 packets/sec

Elapsed time:  0:01
X-exit

paulo@drbl: ~
paulo@drbl: ~
paulo@drbl: ~
IPtraf
Statistics for eth0
-----
                Total      Total      Incoming  Incoming  Outgoing  Outgoing
                Packets    Bytes     Packets   Bytes     Packets   Bytes
Total:          996057    790766K   396241    42624184  599816    748142K
IP:             996057    776422K   396241    36677530  599816    739745K
TCP:           937651    734313K   367004    35682412  570647    698631K
UDP:           58405     42108998  29237     995118   29168     41113880
ICMP:           1         48        0         0         1         48
Other IP:       0         0         0         0         0         0
Non-IP:         0         0         0         0         0         0

Total rates:    14310,5 kbits/sec      Broadcast packets: 22
                3151,6 packets/sec      Broadcast bytes:   4506

Incoming rates: 1578,0 kbits/sec
                1391,0 packets/sec

Outgoing rates: 12732,5 kbits/sec
                1760,6 packets/sec

Elapsed time:  0:01
X-exit

paulo@drbl: ~
paulo@drbl: ~
paulo@drbl: ~
IPtraf
Statistics for eth0
-----
                Total      Total      Incoming  Incoming  Outgoing  Outgoing
                Packets    Bytes     Packets   Bytes     Packets   Bytes
Total:          1103574    890677K   434978    46693397  668596    843983K
IP:             1103574    874827K   434978    40204425  668596    834623K
TCP:           1045127    832714K   405720    39206988  639407    793507K
UDP:           58446     42112813  29258     997437   29188     41115376
ICMP:           1         48        0         0         1         48
Other IP:       0         0         0         0         0         0
Non-IP:         0         0         0         0         0         0

Total rates:    8049,5 kbits/sec      Broadcast packets: 22
                1204,2 packets/sec      Broadcast bytes:   4506

Incoming rates: 979,2 kbits/sec
                487,8 packets/sec

Outgoing rates: 7070,3 kbits/sec
                716,4 packets/sec

Elapsed time:  0:01
X-exit

```