

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**CONSTRUÇÃO DE UM AMBIENTE
VIRTUALIZADO PARA TERMINAIS
THIN CLIENT**

TRABALHO DE GRADUAÇÃO

Ronaldo Canofre Mariano dos Santos

Santa Maria, RS, Brasil

2008

CONSTRUÇÃO DE UM AMBIENTE VIRTUALIZADO PARA TERMINAIS THIN CLIENT

por

Ronaldo Canofre Mariano dos Santos

Trabalho de Graduação apresentado ao Curso de Ciência da Computação
da Universidade Federal de Santa Maria (UFSM, RS), como requisito
parcial para a obtenção do grau de
Bacharel em Ciência da Computação

Orientador: Prof. João Carlos Damasceno Lima

**Trabalho de Graduação N. 258
Santa Maria, RS, Brasil**

2008

**Universidade Federal de Santa Maria
Centro de Tecnologia
Curso de Ciência da Computação**

A Comissão Examinadora, abaixo assinada,
aprova o Trabalho de Graduação

**CONSTRUÇÃO DE UM AMBIENTE VIRTUALIZADO PARA
TERMINAIS THIN CLIENT**

elaborado por
Ronaldo Canofre Mariano dos Santos

como requisito parcial para obtenção do grau de
Bacharel em Ciência da Computação

COMISSÃO EXAMINADORA:

Prof. João Carlos Damasceno Lima
(Presidente/Orientador)

Prof. Antonio Marcos de Oliveira Candia (UFSM)

Prof. Benhur de Oliveira Stein (UFSM)

Santa Maria, 01 de Fevereiro de 2008.

“Tudo leva mais tempo do que todo o tempo que você tem disponível.”
“Nada está tão ruim que não possa piorar.”
— MURPHY

AGRADECIMENTOS

Bueno!

Pelos quatro anos de paciência e compreensão e pelo apoio durante essa reta final ao meu pai e a minha mãe que agüentaram o mal humor desta criatura e a minha noiva que até que foi compreensiva e paciente comigo e praticamente sabe todo o conteúdo do meu TG, pois foi um dos meus corretores ortográficos! Um muito obrigado por tudo, pois vocês não só possibilitaram esse momento como também foram uma motivação para que ele existisse!

Aos colegas que com o passar do tempo tornaram-se amigos! Principalmente ao Linck, Márcio e Elias a quem eu incomodei um monte durante este trabalho e apesar das perguntas simplórias, nunca se negaram a responder!

Aos professores um muito obrigado por tudo, inclusive pela paciência nas revisões de provas.

Ao Google e suas funcionalidades, principalmente a ferramenta de idiomas, que me deu uma mão que é coisa de loco!

Por fim, a todos aqueles que eu incomodei, direta ou indiretamente, e que fizeram parte desta conquista, um quebra costela pra lá de especial!

RESUMO

Trabalho de Graduação
Curso de Ciência da Computação
Universidade Federal de Santa Maria

CONSTRUÇÃO DE UM AMBIENTE VIRTUALIZADO PARA TERMINAIS THIN CLIENT

Autor: Ronaldo Canofre Mariano dos Santos

Orientador: Prof. João Carlos Damasceno Lima

Local e data da defesa: Santa Maria, 01 de Fevereiro de 2008.

A administração e manutenção de redes de computadores é uma tarefa presente praticamente em todos os ambientes de trabalho que envolvam computadores ligados em rede. Juntamente a essa tarefa estão presentes os problemas e preocupações que lhe são peculiares, como a manutenção e subutilização das máquinas, por exemplo. Para solucionar alguns desses problemas, passou a ser utilizada a tecnologia de virtualização, que possibilita a execução de diversos Sistemas Operacionais sobre uma mesma máquina. Assim, unindo essa tecnologia à utilização de *Thin Clients* dotados de um *hardware* básico, adquiridos para o laboratório do Núcleo de Ciência da Computação, o presente trabalho tem como objetivo criar um ambiente que viabilize a utilização destes dispositivos tirando o máximo proveito de suas características através da configuração das ferramentas de gerenciamento disponibilizadas para este equipamento.

Palavras-chave: Administração de Redes, Ambiente virtualizado, Terminais leves, Virtualização.

ABSTRACT

Graduation Work
Undergraduate Program in Computer Science
Federal University of Santa Maria

CONSTRUCTION OF A VIRTUAL ENVIRONMENT FOR THIN CLIENT TERMINALS

Author: Ronaldo Canofre Mariano dos Santos
Advisor: Prof. João Carlos Damasceno Lima

The administration and maintenance of computer networks is an active task on practically all work environment that have anything to do with network wired computers. Alongside with that task we have some problems and worries that are quite unique, like the maintenance and low use of the machines, for example. To solve some of these problems the virtualization technique started to be used, which enables the execution of several Operational Systems on the same machine. That way, uniting this technology and the use of the Thin Clients (with a basic hardware included), acquired by the Computer Science Nucleus laboratory, the present work has as a goal to create an environment that makes possible the use of such devices. This way taking the best out of their own characteristics through the configuration of management tools used for these kind of equipment.

Keywords: Administration Network; Virtual Environment; Thin Client; Virtualization; Informática UFSM.

LISTA DE FIGURAS

Figura 2.1 – Estrutura clássica de um sistema virtualizado.	15
Figura 2.2 – Estrutura hospedeira de um sistema virtualizado.....	16
Figura 2.3 – Interface do VMware ESX Server 3.0.....	18
Figura 2.4 – Arquitetura cliente-servidor de 2 e 3 níveis.....	20
Figura 3.1 – Estrutura física da rede utilizada.....	22
Figura 3.2 – Interface administrativa do SRSS.....	24
Figura 3.3 – Diagrama da conectividade entre o SRC e o Windows (Sun Microsystems, 2007a).	25
Figura 4.1 – Estrutura lógica da rede utilizada.	27
Figura 4.2 – Diagrama de funcionamento da inicialização das DTU's.	29
Figura 4.3 – Diagrama de funcionamento dos terminais.	30
Figura 4.4 – Menu com lista das máquinas disponíveis para autenticação.	34
Figura 4.5 – Topologia lógica do grupo <i>failover</i>	35
Figura 4.6 – Execução do SO Windows em uma janela com resolução 800x600. ...	36
Figura 4.7 – Interface gráfica administrativa do SDM.	38

LISTA DE TABELAS

Tabela 4.1 – Endereço IP das interfaces de rede do servidor srss-200-18.	28
Tabela 4.2 – Distribuição dos Endereços IP	34
Tabela 4.3 – Quadro resumos do trabalho desenvolvido.	38

LISTA DE ABREVIATURAS E SIGLAS

DTU	<i>Desktop Units</i>
LDAP	<i>Lightweight Directory Access Protocol</i>
MAC	<i>Media Access Control</i>
MV	Máquina Virtual
MMV	Monitor de Máquinas Virtuais
NFS	<i>Network File System</i>
NCC	Núcleo de Ciência da Computação
PC	<i>Personal Computer</i> ou Computador Pessoal
RDP	<i>Remote Desktop Protocol</i>
SCW	<i>Sun Ray Connector for Windows</i>
SDM	<i>Sun Ray Desktop Manager</i>
SM	<i>Sun Microsystems</i>
SO	Sistema Operacional
SR	<i>Sun Ray</i>
SRSS	<i>Sun Ray Server Software</i>
UCP	Unidade Central de Processamento
ZFS	<i>Zettabyte File System</i>

SUMÁRIO

1	INTRODUÇÃO	12
2	REVISÃO BIBLIOGRÁFICA	14
2.1	Virtualização	14
2.1.1	Abordagens de virtualização	15
2.2	MMV VMware	16
2.2.1	VMware ESX Server	16
2.3	Terminais	17
2.3.1	<i>Thin Client</i>	19
2.3.2	Funcionamento dos <i>Thin Client</i>	19
3	CARACTERÍSTICAS GERAIS E FERRAMENTAS UTILIZADAS	21
3.1	Visão geral	21
3.2	O ambiente	22
3.2.1	Motivação para um novo ambiente	22
3.3	Conjunto de aplicativos para gerenciamento de terminais Sun Ray	23
3.3.1	<i>Sun Ray Server Software</i>	23
3.3.2	<i>Sun Ray Connector for Windows</i>	24
3.3.3	<i>Sun Ray Desktop Manager</i>	25
4	DESENVOLVIMENTO DO AMBIENTE VIRTUALIZADO	26
4.1	Desenvolvimento do novo ambiente	26
4.2	Aplicação do <i>Sun Ray Server Software</i>	27
4.2.1	O gerenciamento dos <i>Thin Client</i> pelo SRSS	28
4.2.2	Gerenciamento de periféricos	29
4.2.3	Configuração de um grupo <i>failover</i>	33
4.3	Aplicação do <i>Sun Ray Connector for Windows</i>	35
4.4	O <i>Sun Ray Desktop Manager</i>	37
4.5	Quadro resumo do trabalho desenvolvido	38
5	CONCLUSÃO	39
	REFERÊNCIAS	41

1 INTRODUÇÃO

A virtualização de recursos computacionais foi um conceito amplamente difundido na década de 70, visando um melhor aproveitamento do poder dos *mainframes*. Isso através de uma melhor utilização dos recursos disponíveis, possibilitado pela capacidade de execução de diversos Sistemas Operacionais (SO) sobre um mesmo *hardware*, oferecendo ainda economia, flexibilidade, segurança, gerenciabilidade de sistemas de software e um bom isolamento de falhas .

Grande parte dessas características oferecidas pela virtualização são amplamente desejadas e encontram-se presentes no cotidiano da grande maioria das instituições e administradores de redes (TORRES, 2001). Dentre os fatores positivos disponibilizados pela virtualização e almejados pelas instituições, principalmente para laboratórios como o NCC (Núcleo de Ciência da Computação - UFSM), podemos citar a disponibilização de um maior número de computadores atrelada a um baixo investimento.

A execução de diversos SO sobre um mesmo *hardware* também pode ser relacionada com a capacidade de utilização de uma mesma máquina com um certo poder computacional para suprir as necessidades de outras máquinas com uma capacidade muito baixa de processamento, fazendo com que estas se comportem como terminais, podendo assim reutilizar computadores que ficaram obsoletos com a evolução da tecnologia e o aumento dos requisitos mínimos exigidos por programas de computadores.

Essa metodologia também pode ser aplicada aos *Thin Clients*, ou terminais leves, que, assim como máquinas antigas, possuem um baixo poder computacional, dependendo assim de um servidor para que suas operações sejam realizadas. Esses terminais possuem apenas alguns dispositivos básicos de *hardware* e uma pequena imagem de boot para inicializar uma conexão a fim de obter uma área de trabalho remota disponibilizada pelo servidor.

A utilização de Máquinas Virtuais (MV) como tecnologia para servidores é uma tendência em crescimento no mercado atual e juntamente com a utilização de terminais leves, podem gerar inúmeros benefícios tanto para administradores de rede como para os proprietários das mesmas.

Outra tendência que justifica a utilização desses dispositivos é a computação orientada a serviços, proporcionando a realização de tarefas com processamento remoto, onde os servidores realizam a computação do cliente. Como exemplo, podemos citar o Google, que oferece os mais diversos serviços (GOOGLE, 2007), com os pacote *office*, agenda, base de dados, tocador de áudio entre outros, sem a necessidade ter tais aplicativos instalados, permitindo seu acesso em qualquer computador que possua conexão com a Internet.

Devido as suas características, a aplicação de tecnologias de virtualização pode possibilitar um fácil gerenciamento do SO utilizado, uma ampliação dos laboratórios e uma melhoria no aproveitamento da capacidade de processamento das máquinas com poder computacional mais elevado. Esses fatores convergem para uma característica comum visada por todos, a economia, pois a possibilidade de se obter serviços remotos, sem a necessidade de processamento local, diminui a necessidade de novas máquinas completas, dentre outros custos.

Buscando expandir o laboratório do NCC com um baixo recurso disponível, foram adquiridos Terminais Leves da *Sun Microsystems* (SM), a fim de melhor explorar os recursos das máquinas já existentes no laboratório. Assim, este trabalho tem por objetivo utilizar as ferramentas de gerenciamento disponibilizadas pela *Sun Microsystems* para interligar os *Thin Clients* com as máquinas existentes no NCC, explorando todos os recursos disponíveis nestes terminais e facilitar o seu gerenciamento. Como tecnologias complementares foram utilizados o *software* VMware ESX Server 3.0 da empresa VMware (VMware, 2007) e o SO Solaris 10 da *Sun Microsystems* (Sun Microsystems, 2007).

O restante deste trabalho está organizado da seguinte forma. O capítulo 2 apresenta uma revisão da literatura sobre virtualização de recursos computacionais abordando o Monitor de Máquinas Virtuais (MMV) VMware, terminais e os *Thin Clients*. No capítulo 3 apresenta-se uma visão geral do ambiente, as motivações para um novo ambiente e as ferramentas a serem utilizadas. No capítulo 4, é descrita a configuração do ambiente criado, bem como suas funcionalidades, apresentando-se por fim a conclusão do trabalho.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo apresenta-se uma revisão bibliográfica sobre os tópicos relacionados a este trabalho. Primeiramente é feita uma introdução sobre virtualização e suas abordagens, seguindo-se por uma descrição do MMV VMware. Por fim, é realizada uma introdução a cerca de terminais e suas utilizações.

2.1 Virtualização

Sucintamente, virtualização é a capacidade de utilização de recursos computacionais para representação de um conjunto de *software* e *hardware* com arquiteturas diferentes da arquitetura sobre a qual os recursos estão sendo virtualizados. Dessa forma, proporciona-se uma visão lógica dos recursos de computação, possibilitando uma maior flexibilidade, portabilidade e gerenciabilidade de sistemas.

Na década de 70, a IBM abordou o conceito de virtualização com o desenvolvimento dos *mainframes* a fim de obter uma melhor utilização do poder computacional destas máquinas. Além dessa vantagem, a utilização de Máquinas Virtuais (MV), também proporcionou uma melhor segurança, confiabilidade e consolidação dos servidores (GOLDBERG, 1974).

A rápida evolução dos computadores tornou muitas máquinas ultrapassadas rapidamente, a medida que surgiam novas máquinas com um poder de processamento cada vez maior à um custo decrescente. Isso e o fato da crescente interconectividade, fez com que ressurgissem as motivações visando as tecnologias de virtualização. Dessa forma, novas maneiras de se empregar virtualização surgem em um contexto de rápida evolução tecnológica.

Uma delas é o emprego de máquinas obsoletas ou arquiteturas com um *hardware* reduzido, na forma de terminais, conectadas a um ou mais servidores dotados de um

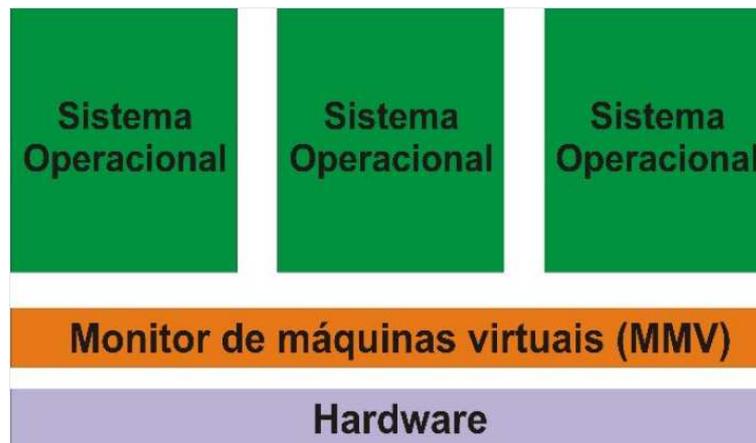


Figura 2.1: Estrutura clássica de um sistema virtualizado.

hipervisor executando diversas máquinas virtuais. Dessa maneira é possível reutilizar os dispositivos obsoletos e obter um melhor aproveitamento do poder de processamento dos servidores, isso a um custo relativamente baixo (WHATELY; AMORIM, 2005).

Uma MV é um código compilado que executa da mesma forma que um computador completo, sendo gerenciado por um MMV. O fato de ser independente de plataforma torna possível a criação de máquinas com diversas configurações e arquiteturas. Isso permite a instalação de SO diferentes executando de maneira independente em uma mesma máquina física.

A responsabilidade sobre as diversas MV executando sobre um mesmo *hardware* é do MMV, ou *hipervisor*, uma camada de *software* que utiliza o conceito de abstração como um passo intermediário para realizar o mapeamento necessário entre o *hardware* real e o virtual. Provendo assim o gerenciamento, escalonamento e alocação dos recursos disponíveis para cada uma das máquinas virtuais criadas, permitindo uma execução transparente e independente. Essa estrutura pode ser vista na figura 2.1.

2.1.1 Abordagens de virtualização

A abordagem utilizada para implementação de um MMV é o fator de classificação do mesmo, levando em consideração o posicionamento do monitor na hierarquia de níveis de execução. Essa classificação envolve sistemas de virtualização do tipo hospedada e sistemas de virtualização do tipo clássica .

Em sistemas de virtualização clássica, o MMV é executado diretamente sobre o *hardware* e as MV's são executadas na camada subsequente. Assim, o monitor executa com maior nível de prioridade, permitindo-se interceptar e avaliar as instruções oriundas das

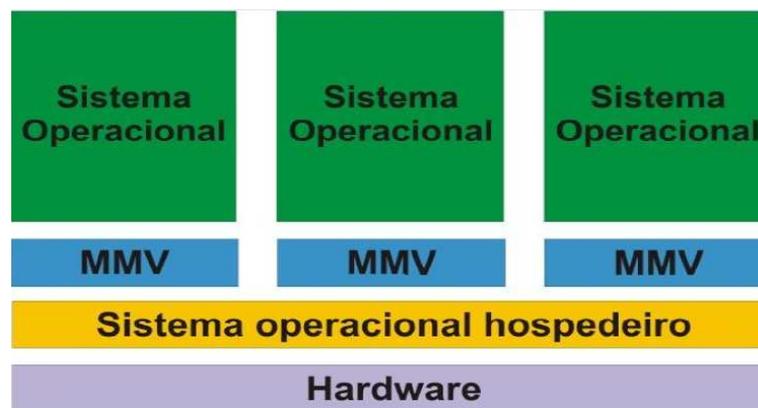


Figura 2.2: Estrutura hospedeira de um sistema virtualizado.

máquinas, direcionado-as para execução direta, sem a necessidade de interpretação. O Xen (BARHAM et al., 2003) é um exemplo de *hipervisor* que adota essa implementação, ilustrada na figura 2.1

Algumas arquiteturas semelhantes à IA-32 não apresentam suporte à virtualização clássica, sendo assim necessárias modificações no kernel do SO. Essa técnica é conhecida com paravirtualização (YOUSEFF et al., 2006). Sistemas que não necessitam dessas modificações são denominados de *Full Virtualization* ou virtualização total.

No caso de sistemas que de virtualização hospedada, a execução da camada de virtualização é realizada sobre um sistema hospedeiro e a MV é instalada como uma aplicação, conforme a figura 2.2. O acesso aos dispositivos físicos é realizado através dos drives de dispositivos do SO e também podem obter outros serviços de sistema sem o auxílio do monitor. Essa arquitetura é adotada por monitores como OpenVZ (OpenVZ, 2007), VMware e o Virtual PC (Microsoft Corporation, 2007) da Microsoft.

2.2 MMV VMware

O VMware é um software de virtualização desenvolvido pela VMware Inc. que apresenta várias versões com propósitos específicos, como VMware Workstation, VMware Player, VMware Server e a versão utilizada nesse trabalho, o VMware ESX Server, assunto que será abordado nas seções a seguir.

2.2.1 VMware ESX Server

O VMware ESX Server, é um SO que utiliza kernel proprietário baseado no SimOS (Stanford University, 2007). Adotando abordagem de virtualização hospedada e

total, executa em um nível próximo ao *hardware*, permitindo assim uma relativa segurança e considerável redução do *overhead*¹ gerado por um sistema base (WIKIPÉDIA, 2007a). Oferece suporte aos sistemas operacionais Windows, Linux, Solaris e Netware, para arquiteturas de 32 e 64 bits.

O ESX Server adota a arquitetura *bare metal*² inserindo uma camada de virtualização diretamente sobre o servidor, facilitando o acesso. A utilização de recursos como processador, memória e banda de rede, é controlada por um sistema denominado "cotas justas", onde um valor fixo é atribuído a cada recurso, para cada máquina, sendo definido também um limite para o mesmo. Dessa forma a virtualização da Unidade Central de Processamentos(UCP) adota um equilíbrio de carga entre os processadores disponíveis, gerenciando assim a execução do processamento das máquinas virtuais. Ainda, caso várias máquinas virtuais estejam executando o mesmo SO elas terão várias páginas de memória idênticas. O esquema de paginação de memória adotado possibilita uma utilização mais eficiente da mesma através do agrupamento destas páginas idênticas.

Quanto à modificação da capacidade dos recursos, o VMware permite que sejam alterados os valores de memória, espaço de disco e banda de rede através da modificação direta ou do acréscimo de outro dispositivo. Referente aos recursos de processamento, cada máquina virtual existente pode ter somente o número de UCPs modificado. Com relação ao controle das conexões de rede, o ESX proporciona a criação de *switchs* virtuais para controle das interfaces de rede de cada máquina que ao serem criadas recebem um endereço MAC (*Media Access Control*) exclusivo. Todo esse gerenciamento do VMware ESX Server 3.0 é possibilitado através de uma interface, que é mostrada na figura 2.3, a qual possibilita ainda a monitoração de desempenho e geração de relatórios.

2.3 Terminais

O crescente número de computadores tanto nas empresas como nas residências, aumenta também a preocupação e os gastos com manutenção, além de não aproveitarem totalmente o potencial da máquina. Existe ainda o fato da presença de computadores em praticamente todos os locais de trabalho, até mesmo para as atividades mais simples. Assim, a maioria das tarefas realizadas em um ambiente de trabalho convencional ou no

¹Tempo em excesso, gasto com qualquer tipo de armazenamento ou processamento de um recurso requerido, podendo piorar o desempenho do sistema que sofreu o *overhead*.

²No contexto de virtualização, é utilizado para classificar o *software* que executa mais próximo do *hardware*, sem intervir na virtualização ou no SO (WIKIPÉDIA, 2007b).

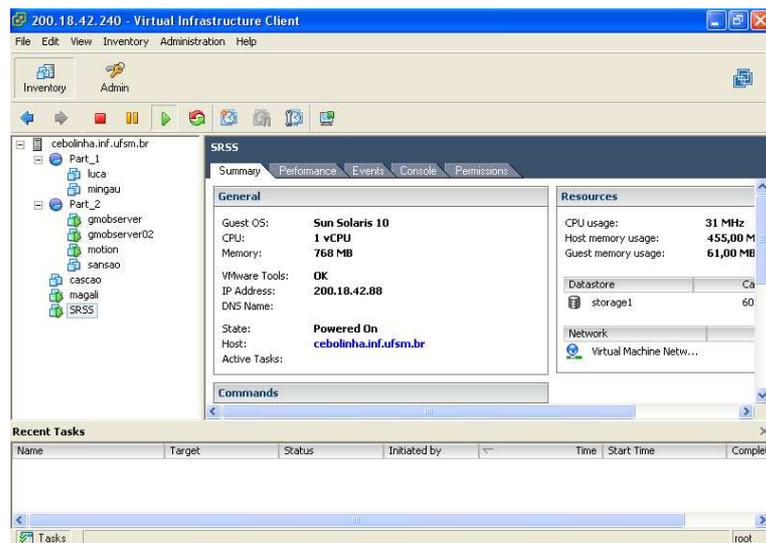


Figura 2.3: Interface do VMware ESX Server 3.0.

uso pessoal do PC, como utilização de um pacote *office*, envio de e-mail, bate papos e pesquisas na *web* por exemplo, não usufruem completamente do poder computacional disponibilizado pela máquina.

Uma alternativa para facilitar as atividade de gerência de redes, diminuir os custos, proporcionar um melhor aproveitamento para o poder de processamento dos computadores atuais e reaproveitar os mais antigos, é a utilização de um microcomputador de maior poder computacional atuando como servidor para terminais.

Terminais são equipamentos com um hardware básico e geralmente simples, sem disco rígido, através dos quais se disponibiliza aos usuários uma interface com um sistema mais complexo. O uso de terminais era muito comum com os *mainframes* proporcionando vários pontos de acesso. Atualmente esse dispositivo pode proporcionar um meio de acesso conveniente e de baixo custo a sistemas dotados de maior poder computacional. Esse acesso pode ser realizado com a utilização de Terminais *dumb* ou Terminais X.

Os terminais *dumb* ou burros, são interfaces com poder computacional limitado, geralmente interagindo com um sistema de informação mais abrangente através de uma conexão serial. São assim chamados pelo fato de sua capacidade de envio e apresentação de informações ser exclusivamente em modo texto e a execução de qualquer programa ser realizada pelo computador em que o terminal está conectado.

Já os terminais denominados X, são mais sofisticado, sendo capazes de exibir aplicações em modo gráfico como as do protocolo *X Window System* (OPENIT, 2007), através de conexões via rede. Os servidores para estes dispositivos podem estar localizados em

uma rede local (*Local Area Network* - LAN) ou numa rede metropolitana (*Metropolitan Area Network* - MAN).

Como os terminais não utilizam o poder de processamento local, máquinas mais antigas e lentas como as que utilizam micro-processadores Pentium II e AMD K6, podem facilmente ser utilizadas da mesma forma que estes equipamentos, dispensando também a utilização de um disco rígido.

2.3.1 *Thin Client*

Thin Clients ou Clientes Magros, são computadores *diskless*³ interligados em rede através de uma arquitetura cliente/servidor, no qual o computador cliente possui poucos ou até mesmo nenhum aplicativo instalado, sendo totalmente dependente do servidor para a execução das aplicações (WIKIPÉDIA, 2007c). O termo *Thin*, refere-se ao fato de estes clientes possuírem um *hardware* mais limitado que uma máquina normal. Tais dispositivos necessitam tipicamente de uma pequena imagem de boot para inicializar uma conexão com a rede e utilizar um navegador *web* dedicado ou uma conexão com uma área de trabalho remoto .

O projeto desses terminais leva em consideração apenas funcionalidades que venham a ser úteis para programas de interface de usuário. O armazenamento não é realizado em discos rígidos impossibilitando que os mesmos sejam corrompidos por um *software* malicioso ou incompatível. Para esse fim são utilizados dispositivos como CD-ROM, memória flash e discos virtuais de rede, garantindo baixos custos de manutenção e o aumento do tempo médio entre falhas (MTBF - *Mean Time Between Failures*) (NOGUEIRA, 2007). A utilização dos *Thin Clients* se dá através dos dispositivos básicos de I/O existentes em uma máquina completa (teclado, *mouse* e monitor).

2.3.2 Funcionamento dos *Thin Client*

Analisando-se uma rede sobre a arquitetura cliente-servidor, o modelo mais habitual, é o de dois níveis, onde o cliente envia uma requisição ao servidor e este, assim que possível, processa a solicitação e envia uma resposta para o cliente. Este modelo está representado na figura 2.4(A).

Neste modelo, uma tarefa requisitada é dividida em duas partes: a apresentação e o processamento realizados pelo cliente (um computador normal) e o acesso aos dados

³O mesmo que **sem disco**.

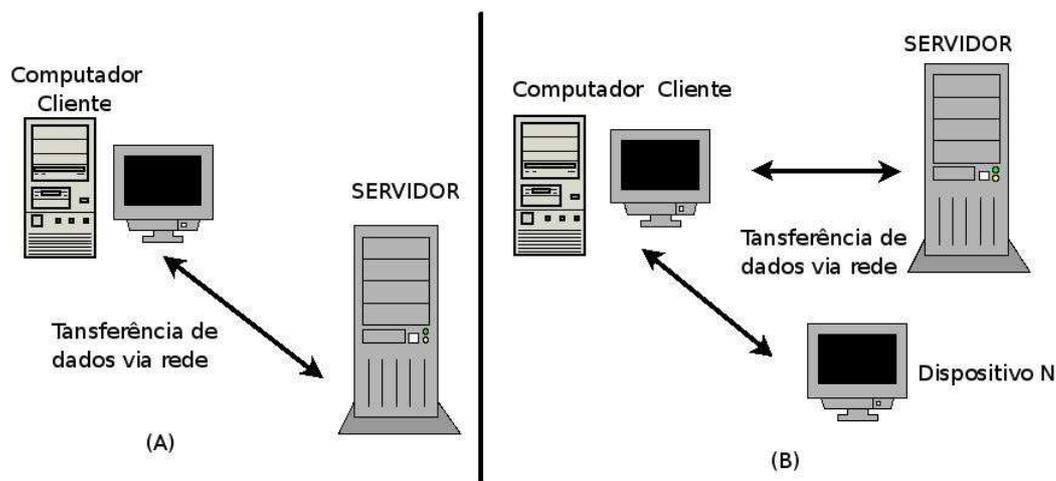


Figura 2.4: Arquitetura cliente-servidor de 2 e 3 níveis.

provido pelo servidor. Como exemplo dessa arquitetura, tem-se a rede do NCC, onde os dados armazenados em servidores são manipulados através dos computadores disponíveis no laboratório.

Entretanto, buscando a melhoria do modelo anterior, desenvolveu-se uma arquitetura de 3 níveis, representada na figura 2.4(B). Essa arquitetura caracteriza-se por dividir as tarefas em 3 partes, onde a disponibilização dos dados continua sendo realizada pelo servidor, ao passo que a estação cliente passa a realizar somente o processamento dos dados, ficando a exibição dos resultados sob a responsabilidade de um outro dispositivo, um terminal por exemplo.

O funcionamento da arquitetura *Thin Client*, pode ser implementado acessando diretamente um servidor que realiza o processamento e acesso aos dados, ficando o terminal responsável apenas pela apresentação dos resultados. No entanto ele também pode ser configurado de acordo com a utilização da arquitetura cliente-servidor de 3 níveis, onde a estação cliente processa as tarefas enviadas pelo terminal, e este disponibiliza os resultados na tela. Em ambas as implementações, o terminal fica responsável apenas pelo processamento da camada de apresentação, ou seja, realiza apenas uma atualização do *display* da estação (SOUZA FILHO, 2003).

3 CARACTERÍSTICAS GERAIS E FERRAMENTAS UTILIZADAS

Neste capítulo é exposta uma visão geral do ambiente que se deseja configurar expondo os motivos e necessidades para tal, bem como as vantagens de sua utilização. Por fim, é abordada uma visão geral sobre as ferramentas a serem utilizadas para o novo ambiente.

3.1 Visão geral

Inicialmente a configuração desse ambiente tinha como objetivo ser aplicada a 10 terminais Sun Ray 1G da *Sun Mycrosystems* visando a ampliação do laboratório do NCC. Estes terminais disponibilizam conexão *Ethernet*, através da qual é realizada a troca de dados com os servidores de ambiente gráfico, placa de vídeo, entrada e saída de áudio, dispositivo para leitura de *smart card* (cartão inteligente) e 4 portas USB, através das quais são conectados também teclado e *mouse*.

O ambiente a ser configurado visa o funcionamento de grande parte desses dispositivos dos quais as estações são dotadas, tendo como servidores de ambiente gráfico as máquinas Soyo do NCC executando o SO Gentoo, podendo ainda utilizar outras MV que sejam disponibilizadas. Para o SO Windows, os *Thin Client* utilizarão máquinas do NCC, já configuradas para disponibilizar acesso remoto a este sistema.

Juntamente com o ambiente configurado, objetiva-se também a disponibilização de uma interface administrativa *web* para gerenciamento desses terminais. Todos os objetivos propostos pretendem ser alcançados utilizando-se o SO Solaris 10 e as ferramentas de gerenciamento para as Sun Ray (SR), também chamadas de DTUs (*Desktop Units*), ambos disponibilizados pela SM. A seguir é descrito o ambiente anterior e as motivações para substituição do mesmo.

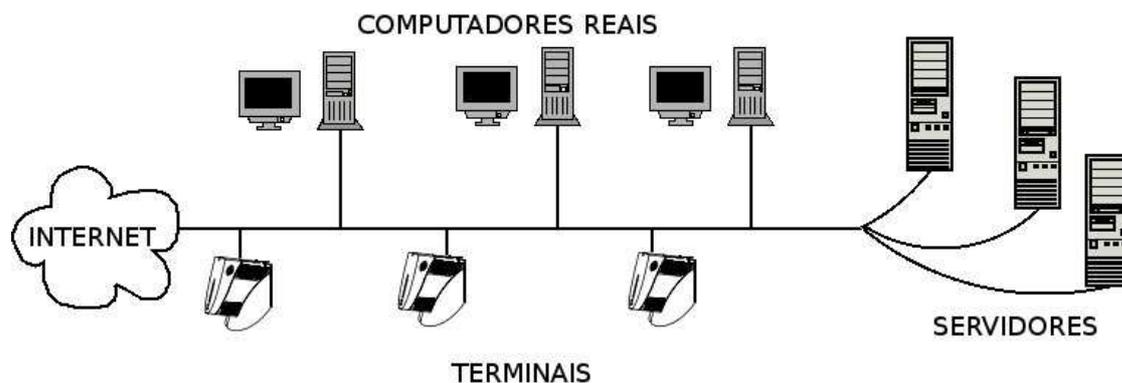


Figura 3.1: Estrutura física da rede utilizada.

3.2 O ambiente

Anteriormente os terminais SR em funcionamento no laboratório do NCC eram gerenciados por uma versão, para o SO Linux, do *Sun Ray Server Software* (SRSS), que basicamente é responsável pelo gerenciamento e descoberta dos recursos de rede.

Embora a utilização de terminais possa ser realizada conectando os mesmos diretamente a servidores, a estrutura da rede física sobre a qual se desenvolveu este trabalho, segue a estrutura da figura 3.1, onde é possível observar que todos os dispositivos envolvidos, terminais, servidores e máquinas anfitriãs, encontram-se sobre um mesmo barramento.

A distribuição Linux utilizada anteriormente era o Debian versão 4.1.1-21, kernel 2.6.18 em uma máquina virtual com dois processadores AMD Opteron de 2.7 GHz cada, tendo 893 MHz como frequência reservada para cada um dos dois. A configuração de memória garantia um mínimo de 837 MB até o limite máximo disponível. O espaço de disco reservado era de 10 GB e a mesma possuía três adaptadores de rede.

O ambiente atual disponibiliza acesso ao SO Gentoo versão 3.4.6-r1, kernel 2.6.16 e o *hardware* compartilhado tem como características processador de 2.0 GHz e 512 KB de memória em cada máquina servidor, não disponibilizando acesso a todos os dispositivos das DTU, como áudio e *pendrive*. Atualmente também não está disponível a utilização do SO Windows, que necessita do *Sun Ray Connector for Windows* (SRC).

3.2.1 Motivação para um novo ambiente

A constante necessidade de atualização de programas e instalação de *software* bem como a necessidade de manutenção nas máquinas presentes no laboratório do NCC da

UFSM, consome um bom tempo dos bolsistas responsáveis pela manutenção do mesmo. Esse fato ocorre também em muitas outras instituições e empresas (PC&CIA, 2007), bem como em outros centros da própria UFSM.

Quanto à utilização do laboratório, pode-se afirmar que este encontra-se frequentemente repleto de alunos que o utilizam para realização dos trabalhos acadêmicos. Também, não raramente, são oferecidos míni-cursos, oficinas e até mesmo aulas que necessitam da utilização dos microcomputadores do laboratório ou da sala informatizada Professor Cláudio Rocha Lobato.

Unindo-se a essas constatações, o fato da constante e vertiginosa evolução das tecnologias e do poder de processamento dos componentes de *hardware*, a melhoria do laboratório do curso, assim com as melhorias em outras instituições públicas ou privadas, tem como gargalo o alto custo a ser empregado.

Todos esses fatores podem ser facilmente contornados pela utilização de *Thin Client* pois, conforme já explicado anteriormente, além de ampliar o laboratório do NCC com qualidade e melhor aproveitar o poder de processamento das máquinas existentes o mesmo pode ainda proporcionar:

- menor custo de administração e implantação;
- facilidade de proteção e
- baixo consumo de energia.

3.3 Conjunto de aplicativos para gerenciamento de terminais Sun Ray

Para a utilização do modelo de computação para qual o *Thin Client* foi desenvolvido, a Sun disponibiliza um conjunto de aplicativos que facilita e possibilita uma utilização completa dos recursos das DTU's. A seguir serão descritas essas ferramentas, as quais foram utilizadas durante este projeto, para a configuração do ambiente.

3.3.1 Sun Ray Server Software

O *Sun Ray Server Software* (SRSS) é o principal componente, tendo sido utilizado para as configurações do ambiente dos *Thin Clients*. Atualmente na versão 4.0, está disponível somente para Solaris, SuSE Linux Enterprise Server e Red Hat Enterprise Linux

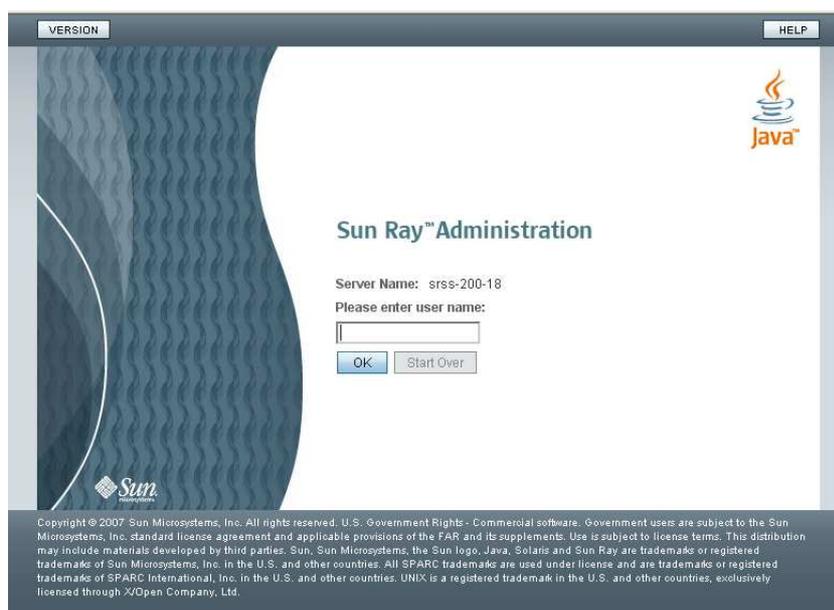


Figura 3.2: Interface administrativa do SRSS.

Advanced Server. Essa ferramenta é responsável pela criação e administração da rede de terminais, realizando a conexão entre os terminais e os computadores anfitriões, configuração e controle de tipos de acesso, definição de propriedades, monitoramento do sistema e solução de problemas, dentre outras.

A realização dessas tarefas pode ser feita por linha de comando ou por meio da interface administrativa *Graphical User Interface* (GUI) que utiliza *Java Server Faces* (JSF), para processar dinamicamente uma interface baseada na *web*, que pode ser vista na figura 3.2. A exibição desta interface pode ser configurada para permitir acesso somente a partir da rede interna (configuração padrão) ou habilitado para acesso pela Internet.

3.3.2 *Sun Ray Connector for Windows*

A utilização de servidores remotos para executar o SO Windows pelos *Thin Clients* é possibilitada pelo *Sun Ray Connector for Windows*, um cliente de serviços de terminal com suporte da Sun e baseado no RDP (*Remote Desktop Protocol*) da Microsoft. Proporciona ainda um acesso através da utilização completa da tela ou pela execução em uma janela do ambiente Solaris ou Linux (Sun Microsystems, 2007a).

De uma forma geral, o SCW pode ser tratado com um intermediário entre o terminal e o servidor de terminais Windows. Além da utilização do RDP para comunicação com o servidor, utiliza também o protocolo ALP (*Appliance Link Protocol*) para realizar a comunicação com a DTU, como mostra a figura 3.3.

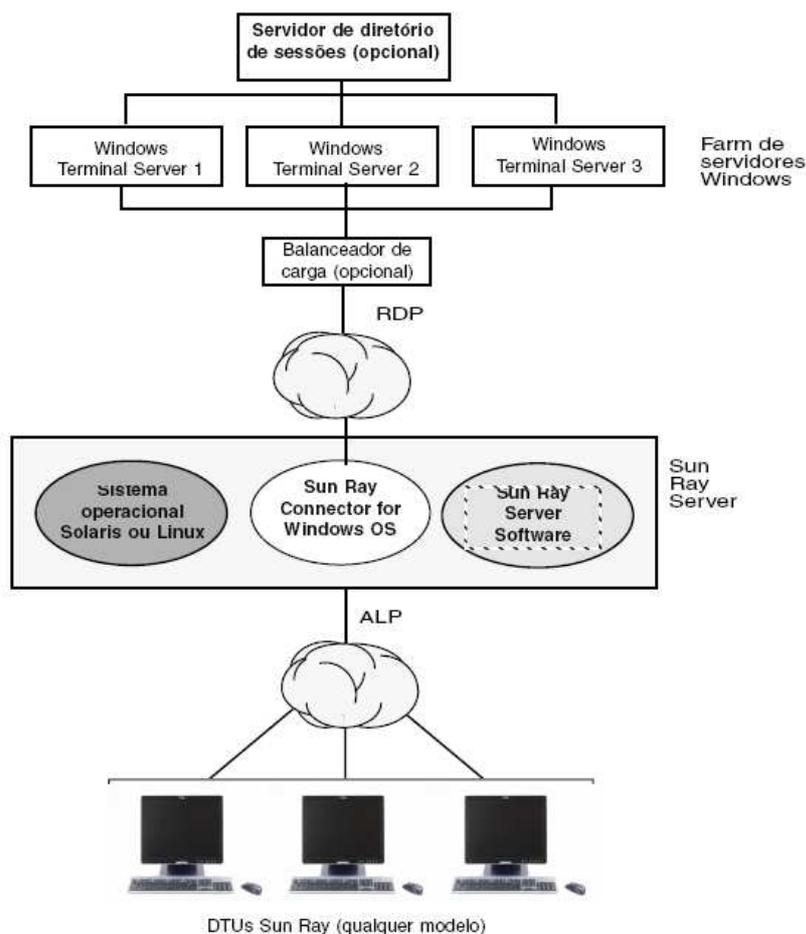


Figura 3.3: Diagrama da conectividade entre o SRC e o Windows (Sun Microsystems, 2007a).

3.3.3 *Sun Ray Desktop Manager*

O *Sun Ray Desktop Manager* (SDM) é a ferramenta disponibilizada pela Sun, que visa facilitar e centralizar a configuração para computadores anfitriões. Basicamente ela fornece uma estrutura para armazenar configurações para aplicações em uma rede, de forma centralizada para os usuários e de máquinas que executam o pedido.

O SDM visa possibilitar ferramentas de configuração de repositório, permitindo armazenar as configurações de perfil e as estruturas organizacionais, disponibilizando um gerenciamento realizado pela interface administrativa ou por linha de comando e também configuração de agentes e adaptadores, que armazenam as configurações de usuário nos repositórios aplicando-as posteriormente. As configurações que não são disponibilizadas por padrão podem ser criadas ou adaptadas pelo administrador, através da inserção de modelos (Sun Microsystems, 2007b).

4 DESENVOLVIMENTO DO AMBIENTE VIRTUALIZADO

Este capítulo apresenta o ambiente desenvolvido neste trabalho, sendo abordadas de forma mais detalhada cada uma das ferramentas utilizadas, suas funcionalidades e funcionamento.

4.1 Desenvolvimento do novo ambiente

Como servidor para o novo ambiente foi utilizada uma máquina virtual cujas características diferem da anterior em três aspectos: conta com processador de 2.0 GHz, disco rígido de 8 GB e sistema operacional Solaris 10. Sistema este desenvolvido pela *Sun Microsystems*, que é responsável também pelo desenvolvimento dos *Thin Client* 1G sobre os quais se desenvolveu este trabalho, bem como pelos *softwares* utilizados para configuração dos terminais.

O fato de os produtos utilizados serem desenvolvidos pela mesma empresa que o Solaris, foi um dos itens relevantes na escolha do mesmo, pois embora sejam disponibilizados *softwares* para algumas das distribuições Linux, eles são desenvolvidos e melhor suportados para o SO da SM. Dessa forma seria possível também obter um maior número de ferramentas administrativas disponibilizadas para o SO Solaris e uma maior prioridade na obtenção de atualizações e correções, visto que as mesmas são realizadas primeiramente para o sistema da SM.

Assim, buscando contar com um maior suporte e a possibilidade de pôr em funcionamento todos os dispositivos e características oferecidas por um *Thin Client*, este trabalho teve início com a criação da MV, instalação e configuração do sistema escolhido e instalação e configuração do SRSS. A máquina virtual criada foi adicionada à rede principal do NCC através de sua interface principal (pcn0), as outras duas placas restantes

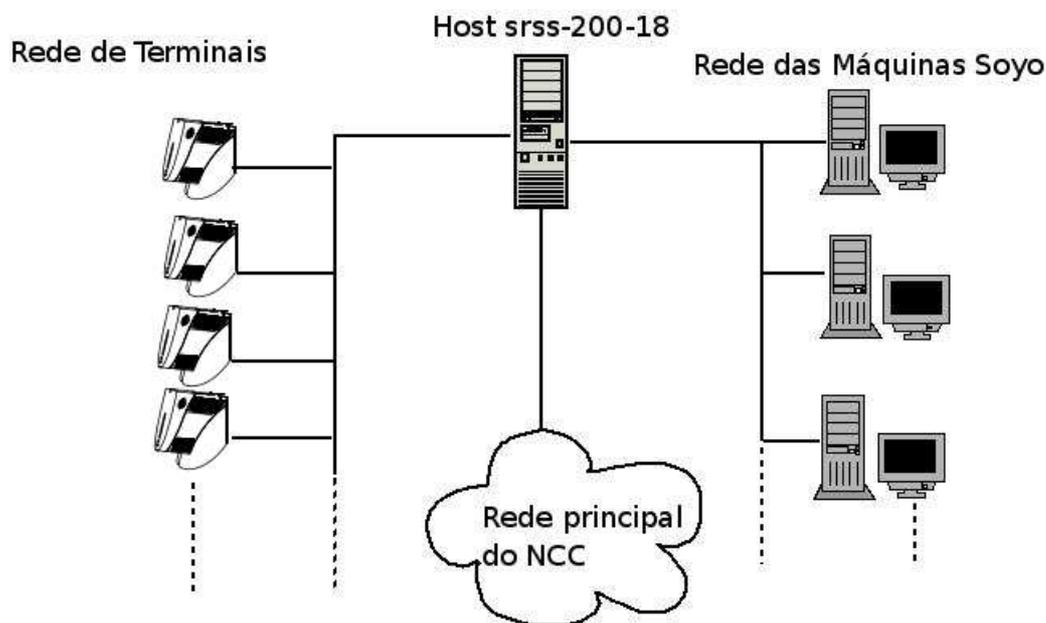


Figura 4.1: Estrutura lógica da rede utilizada.

(pcn1,pcn2) foram conectadas à rede das máquinas anfitriãs (Soyo) e as DTU's, respectivamente. Dessa forma, a topologia geral da rede lógica fica dividida em três, uma para cada grupo de máquinas, como pode ser visto na figura 4.1.

4.2 Aplicação do *Sun Ray Server Software*

Para disponibilizar o funcionamento dos terminais da Sun, o primeiro *software* instalado foi o *Sun Ray Server Software* que é responsável por gerenciar a inicialização e o envio e recebimento de requisições dos terminais para os anfitriões, dentre outras funcionalidades. Através do SRSS 4.0, é possível também que um usuário acesse qualquer uma das sessões em qualquer um dos terminais (Sun Microsystems, 2007c).

Anteriormente à configuração do SRSS, foi necessário configurar uma interface de rede para o mesmo, através do próprio *software*, definindo neste instante as características de rede e o número de DTU's pertencentes a rede, limitando o primeiro e o último endereço de rede, para uma distribuição de endereços dinâmica.

O servidor DHCP (*Dynamic Host Configuration Protocol*) padrão criado durante a instalação foi configurado fazendo com que a distribuição dos endereços IP fosse realizada de forma estática, evitando assim problemas no funcionamento e segurança da rede geral do NCC. Assim, os endereços atribuídos para cada terminal, encontram-se na faixa 10.10.2.X, diferente da faixa utilizada pela rede principal do NCC (200.18.42.X), e da

rede interna do laboratório (10.1.1.X), e são atribuídos somente a máquinas cujo MAC esteja cadastrado.

Diferentemente dos servidores Linux, o SO Solaris não permite a configuração e atribuição do DHCP através da criação e modificação de arquivos, a mesma é feita através de comandos disponibilizados juntamente com a instalação dos pacotes responsáveis pelo serviço. Dessa forma, visando facilitar a manutenção da rede das *Sun Ray*, foram reunidos em um *script*¹, os comandos e opções para tal tarefa, proporcionando uma maior facilidade ao realizar operações de inclusão, remoção, alteração e exclusão de 1 ou n terminais a tabela de endereços do servidor de IP.

4.2.1 O gerenciamento dos *Thin Client* pelo SRSS

Após instalado e configurado, o *software* em questão gerencia todos os terminais cadastrados e conectados a ele, bem como quais computadores anfitriões estão disponíveis, isso através das conexões de rede que envolvem todas as máquinas. Ao ser inicializado, o SRSS percorre todas as máquinas das redes a que estiver conectado, disponibilizando para as DTU's uma listagem daquelas que estão configuradas para exportar seu ambiente X.

Na configuração aplicada, a placa de rede pcn1 recebe um endereço IP na mesma faixa que os terminais, sendo assim responsável pela comunicação entre eles e o servidor. Para captura das máquinas anfitriãs, as placas de rede pcn2 e pcn0 recebem endereços das redes 10 e 200 do NCC, conforme a tabela 4.1, disponibilizando acesso as máquinas anfitriãs encontradas nestas redes.

Interface de Rede	Endereço IP	Rede
pcn0	200.18.42.88	Rede externa (rede 10)
pcn1	10.10.2.1	Rede dos terminais
pcn2	10.1.1.209	Rede interna (rede 200)

Tabela 4.1: Endereço IP das interfaces de rede do servidor srss-200-18.

Quando uma DTU é inicializada, ela estabelece uma conexão com o servidor (*hostname*² srss-200-18), através de um *firmware*³, que envia a requisição ao servidor que por sua vez lhe devolve uma lista das máquinas anfitriãs previamente obtidas pela comunicação do servidor com as redes da qual faz parte, conforme a figura 4.2. Assim o terminal

¹Arquivo do tipo texto contendo comandos que possam ser executados

²Nome que identifica uma máquina

³Um tipo de software armazenado permanentemente em um chip, que controla diretamente o hardware.

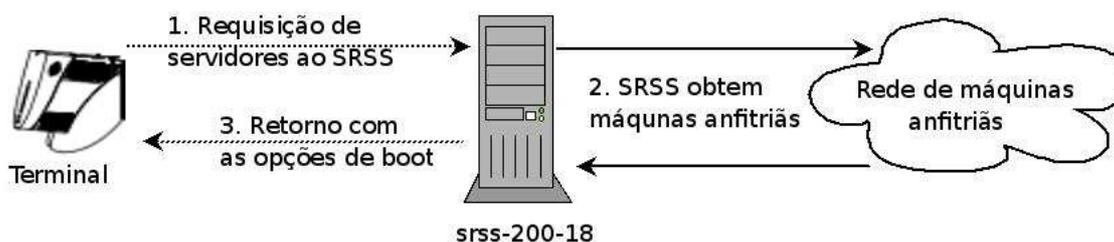


Figura 4.2: Diagrama de funcionamento da inicialização das DTU's.

exibe na tela uma lista com os dados das máquinas disponíveis para o usuário realizar a sua autenticação

Conforme pode ser visto na figura 4.3, a partir do momento que o usuário escolhe a máquina anfitriã na qual vai se conectar, a DTU estabelece uma conexão com o servidor informando os dados desta conexão, solicitando recursos (1). O srss-200-18, estabelece a comunicação entre a DTU e seu anfitrião, enviando para este as requisições do terminal (2). Após realizadas as tarefas solicitadas, o anfitrião envia para o servidor o resultado obtido (3) e este por sua vez encaminha para o terminal para que seja mostrado na tela (4). Assim todas as tarefas e processamentos solicitados pelo usuário no terminal são encaminhadas ao servidor e deste para a máquina anfitriã para que realize o processamento da mesma forma que seria realizado se o usuário estivesse diretamente logado no servidor.

Com relação ao gerenciamento dos dispositivos das DTU's, o SRSS cria um diretório temporário para cada terminal no momento de sua autenticação. Esse diretório é criado no servidor temporariamente e sempre em `/tmp/SUNWut/units/`, sendo mantido durante o tempo que a mesma estiver conectada ao servidor. A identificação do mesmo consiste de um prefixo seguido do endereço MAC da DTU autenticada. Tais diretórios tem como objetivo disponibilizar os diretórios `/dev` e `/device` similares aos do SO Solaris, onde é armazenada uma representação da topologia lógica dos dispositivos conectados ao terminal correspondente (Sun Microsystems, 2007d).

4.2.2 Gerenciamento de periféricos

Dentre os fatores que levaram a configuração de um novo ambiente, encontravam-se a disponibilização do funcionamento dos dispositivos de armazenamento em massa, com *pendrive* principalmente e a utilização do dispositivo de áudio.

Ao conectarmos um *pendrive* em um terminais, um *script* próprio do *software* de gerenciamento, realiza as verificações montando automaticamente o dispositivo em um

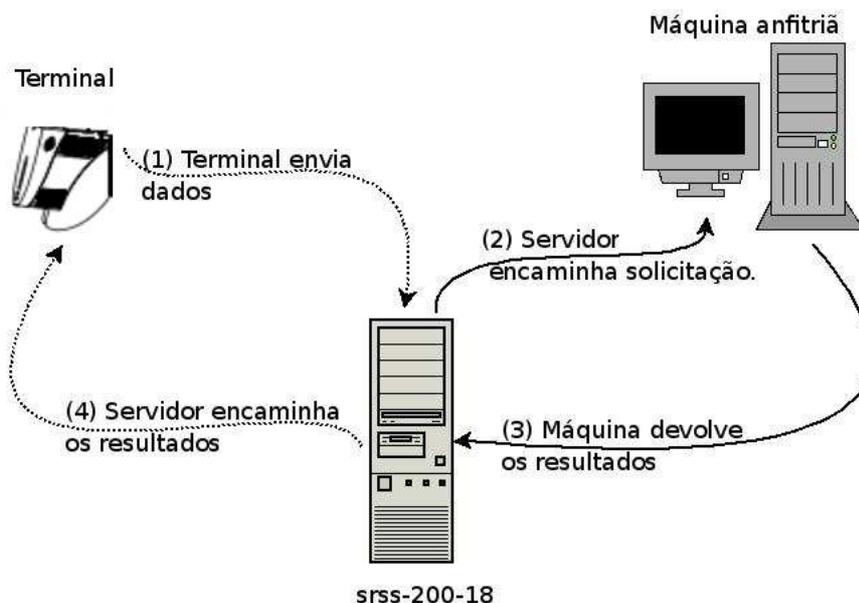


Figura 4.3: Diagrama de funcionamento dos terminais.

diretório nomeado de acordo com o login e localizado em `/tmp/SUNWut/mnt/`. Tais verificações envolvem o sistema operacional utilizado e as informações de usuário. Para o SO Solaris, o dispositivo é montado em um diretório nomeado com o login do usuário, respeitando suas permissões de acesso. Se o sistema operacional em uso for o Linux, o dispositivo é montado para acesso apenas do administrador do sistema em um diretório nomeado como `root` e com permissões de leitura e escrita somente para o administrador.

Buscando soluções para a utilização do dispositivo *pendrive* através dos terminais, foram analisadas e testadas várias possibilidades, dentre elas:

- exportação direta dos dispositivos: exportar os dispositivos montados no SRSS e incluir os diretórios nos arquivos `/etc/fstab` de modo que fosse possível ao usuário acessar o dispositivo, analogamente ao que é feito com a utilização do *pendrive* nas máquinas Soyo. O inconveniente desse método era a falta de segurança quanto o acesso aos dados, pois os mesmos estavam acessíveis a todos os usuários.
- criação de usuários locais: foram cadastrados no servidor, usuários com os mesmos dados (login, uid e gid⁴), de suas contas no NCC, buscando verificar se durante a montagem dos dispositivos, os usuários seriam reconhecidos, dessa forma bastaria realizar a autenticação do servidor também pelo serviço LDAP (*Lightweight*

⁴UID é o número de identificação do usuário para o sistema e GID o número de identificação do grupo ao qual esse usuário pertence

Directory Access Protocol) do NCC.

- alterações no *script* do SRSS: visto que o *script* responsável por montar os dispositivos, é passível de modificações, foram realizadas algumas alterações buscando obter dados do usuário logado, mesmo que este não fosse um usuário Solaris, com a finalidade de modificar o local de montagem para um diretório personalizado. Dessa forma seria possível alterar as permissões, deixando o acesso restrito aos usuários, isso com a utilização de outros fatores e informações, como *uid* e *gid*.

Dessa forma, buscando disponibilizar um acesso seguro aos dispositivos de armazenamento em massa, foi trabalhado com o relacionamento de informações. O servidor *srss-200-18*, armazena várias informações referentes aos terminais conectados a ele, como endereço IP, identificador da sessão e MAC do terminal. Assim, foram verificadas quais destas informações poderiam relacionar o terminal em uso com o dispositivo conectado a ele e o seu respectivo usuário, de forma que o mesmo fosse montado e tivesse acesso permitido somente para o respectivo terminal e proprietário.

Para possibilitar esse acesso, a cada *pendrive* conectado em um terminal, foi necessário verificar em qual DTU o dispositivo se encontrava, e após a sua montagem no servidor, exporta-lo com a utilização do sistema de arquivos *Network File System* (NFS), alterando as permissões do dispositivo, seu proprietário e grupo. Posteriormente a isso, na máquina anfitriã da sessão correspondente, é acrescentada uma linha no arquivo */etc/fstab*, para possibilitar o acesso ao *pendrive*. Esse acesso foi configurado com o auxílio de *script* que são descritos a seguir e possibilitam um melhor entendimento do funcionamento resumido acima.

1. Primeiramente, nas máquinas anfitriãs, um *script* executa a cada autenticação de usuário, atualizando uma lista de usuários no servidor (lista A), contendo os dados referentes ao usuário e a sessão (*login*, *uid*, *grupo*, *gid* e *id* da sessão).
2. Ao conectar um *pendrive* no terminal, o mesmo é montado automaticamente pelo SRSS, e em seqüência são executados comandos que atualizam uma segunda lista (lista B), contendo o nome do dispositivo montado a o seu respectivo terminal. Concatenando as informações das listas A e B, o dispositivo tem seu proprietário e grupo alterado, sendo por fim exportado. Ainda neste momento, antes de exportar

o dispositivo, são removidos os dispositivos que não estão mais disponíveis para acesso.

3. Retornando agora a máquina anfitriã, a cada intervalo de tempo é executada uma seqüência de comandos que verifica a lista B, a procura de dispositivos para a sua sessão, caso encontre, acrescenta uma linha no arquivo */etc/fstab*, permitindo assim a montagem e acesso ao *pendrive*, no diretório */drivers*, da mesma forma que ocorre nas máquinas Soyo. Ainda, anteriormente a essa inclusão, são procurados por dispositivos obsoletos ou pelo mesmo objetivo a ser incluído. No primeiro caso estes são removidos, no segundo, nenhuma ação é realizada.
4. Por fim, novamente nas máquinas anfitriãs, no momento em que o usuário finaliza sua sessão, os dados acrescidos na lista A, são removidos.

Dessa forma torna-se possível obter um acesso seguro aos dispositivos de armazenamento em massa, respeitando as políticas de segurança e acesso implementadas pela administração da rede do NCC e sem a necessidade de grandes modificações e utilização de privilégios de administrador para tal. No entanto restaram ainda algumas outras possibilidades para disponibilização deste acesso, dentre elas:

- criação de vários servidores: criação de um servidor para cada terminal, que poderia solucionar o problema pois cada dispositivo seria montado em um único servidor, no entanto esta possibilidade envolveria um gasto elevado de recursos de *hardware*.
- modificações na estrutura da rede: devido a estrutura da rede do NCC, que visando a sua segurança, assim com em toda rede corporativa, limita o acesso a certas partes de sua estrutura aos administradores da mesma, algumas possibilidades foram logo descartadas, como a montagem remota, a qual os usuários normais não tem permissão de acesso. Dessa forma, a criação de uma MV para cada usuário eliminaria os problemas de permissões o que poderia facilitar o acesso seguro aos dispositivos.
- criação de usuários locais no servidor: para usuários do SO Solaris, a montagem dos dispositivos é feita de forma segura e com acesso restrito. Assim, a criação de um usuário para cada terminal existente, e a autenticação automática deste no servidor, no momento da autenticação do usuário normal, poderia também permitir um acesso fácil e seguro.

Ainda como periférico a ser configurado, restou a disponibilização do dispositivo de áudio nos terminais. Porém, devido a prioridade empregada na disponibilização de acesso aos dispositivos de armazenamento em massa e a tempo empregado para tal, dentre outros fatores, a configuração desta funcionalidade ficou impossibilitada para funcionamento nos terminais ao executarem outro SO diferente do Solaris.

No entanto foram levantadas também algumas possibilidades para tal, dentre elas a criação de um servidor de som para cada terminal conectado. Pois para o funcionamento do áudios, as DTU's necessitariam utilizar o mesmo drive utilizado pelo SO Solaris, no qual os dispositivos de entrada e saída de áudio funcionam corretamente.

4.2.3 Configuração de um grupo *failover*

A utilização de uma arquitetura de rede cliente-servidor, deixa muitos de seus clientes sem acesso quando ocorre algum tipo de falha com um ou mais servidores. Tais problemas podem ocorrer tanto por defeitos em *software* (falhas ou programas maliciosos), quanto por defeitos em *hardware* ou na infra-estrutura onde o servidor se encontra.

Para evitar a indisponibilidade de um servidor de terminais, que poderia ser causada por exemplo, devido a uma queda de energia, problemas na rede ou outra interferência e também para realizar um balanceamento de carga, distribuindo a autenticação e comunicação dos mesmos. O SRSS permite a coexistência com outro(s) servidores do mesmo tipo formando um grupo de falhas ou *failover group*. Isso a fim de evitar que os terminais fiquem inacessíveis, sem possibilidade de autenticação ou muito lentos por ocasião de várias autenticações ou comunicações simultâneas.

Assim, buscando uma maior disponibilidade para a rede dos *Thin Clients*, foi configurado um grupo *failover* com um segundo servidor virtual com as mesmas característica do primeiro servidor. Dessa fora, seguindo o nome dado ao primeiro servidor configurado (srss-200-18) foi atribuído para o segundo servidor a ser configurado o *hostname* srss-200-18-2.

O grupo criado consiste em duas máquinas configuradas para fornecer o serviço de autenticação e gerenciamento das DTU's, tendo um servidor primário e um secundário. Para distribuição de endereço IP. Os servidores foram configurados em duas faixas diferentes, permitindo sua coexistência e funcionamento do grupo tolerante a falhas, os endereços IP ficaram divididos de acordo com a tabela 4.2, dentro da rede 10.10.2.x e a rede lógica

dessa hierarquia pode ser vista na figura 4.5, e consta basicamente de mais uma máquina interligada da mesma forma que o servidor principal.

Hostname	Endereço Servidor	Endereço Inicial	Endereço Final
srss-200-18	10.10.2.1	10.10.2.16	10.10.2.115
srss-200-18-200	10.10.2.2	10.10.2.116	10.10.2.215

Tabela 4.2: Distribuição dos Endereços IP

Basicamente, com este grupo configurado, os terminais autenticam no primeiro servidor que responder a sua solicitação e todos os terminais apresentam a mesma lista de máquinas anfitriãs disponíveis, independente do servidor onde foi realizada a autenticação. Esse funcionamento permite a realização do gerenciamento de carga, distribuindo as solicitações entre os servidores, o que é realizado através de um algoritmo de balanceamento de carga do SRSS. O menu do ambiente desenvolvido, com as máquinas anfitriãs disponíveis para uso, pode ser visto na figura 4.4.

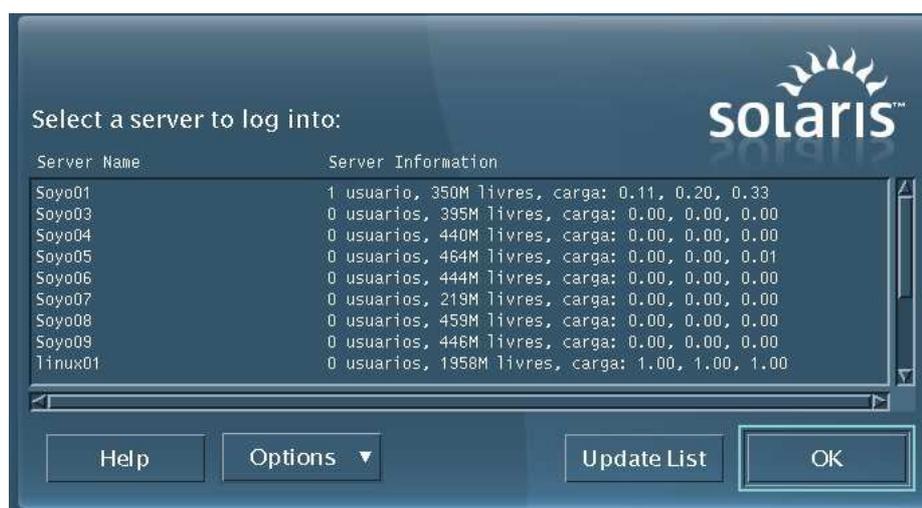


Figura 4.4: Menu com lista das máquinas disponíveis para autenticação.

A maior disponibilidade do ambiente é garantida pela existência dos dois servidores, pois além da divisão da autenticação dos *Thin Clients*, eles funcionam como servidores secundários entre si, ou seja, caso o srss-200-18 fique indisponível, todos os terminais passam a autenticar no srss-200-18-2. Caso ocorra o contrário, e o srss-200-18-2 torne-se indisponível, os terminais passam a se autenticar no srss-200-18.

Um item interessante que pode ser notado é que estes servidores foram denominados como primário e secundário, mas conforme a explicação do funcionamento de ambos no grupo de falhas, podemos ver que eles não trabalham somente desta forma, pois o servidor

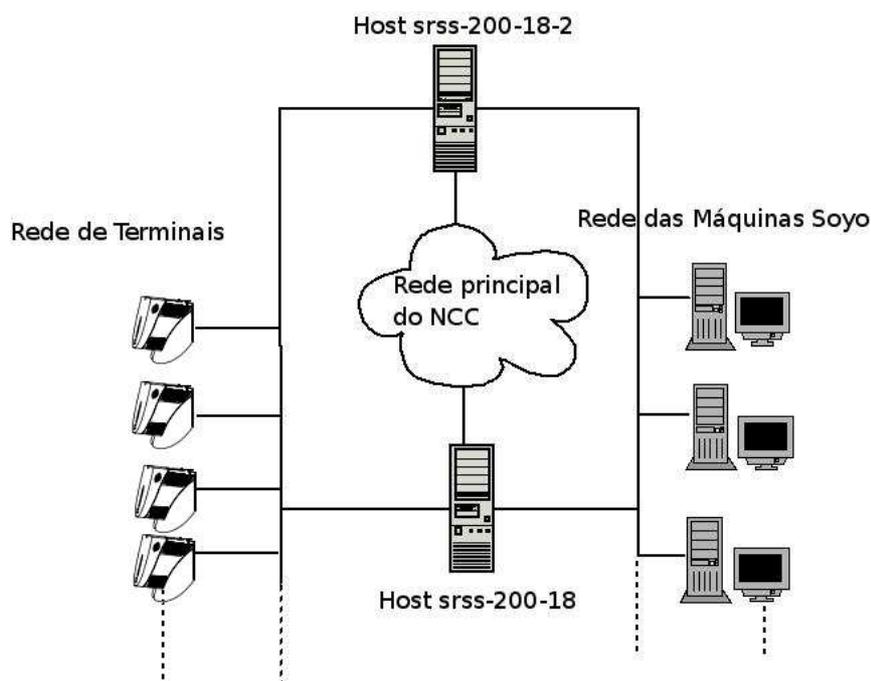


Figura 4.5: Topologia lógica do grupo *failover*.

secundário não é solicitado somente após o servidor primário não responder a solicitação. Eles atuam como servidores primários coexistentes e a medida que um deles falha o outro assume o controle do ambiente, realizando também o balanceamento de carga.

4.3 Aplicação do *Sun Ray Connector for Windows*

A utilização de dois ou mais sistemas operacionais é uma característica fácil de ser encontrada em um ambiente acadêmico. Quer pela diversidade das preferências dos usuários ou pela própria necessidade gerada pelo mercado de trabalho.

Muitas vezes faz-se necessária a utilização de programas e ferramentas específicas ou com um melhor desempenho para um determinado sistema operacional, com por exemplo a ferramenta *Corel Draw*. Existe também a necessidade de realização de trabalhos ou projetos que independam do sistema utilizado, o que faz com que seja necessário possibilitar a utilização de outro SO além do Linux, atualmente disponibilizado pelas máquinas Soyo.

Dentre os servidores do Núcleo de Ciência da Computação, encontram-se algumas máquinas que exportam o ambiente do sistema operacional Windows XP, utilizado na sala Professor Cláudio Rocha Lobato, através das máquinas que atuam como terminais. Desta forma, por já possuir um sistema disponível para ser utilizado e também pela *Sun*

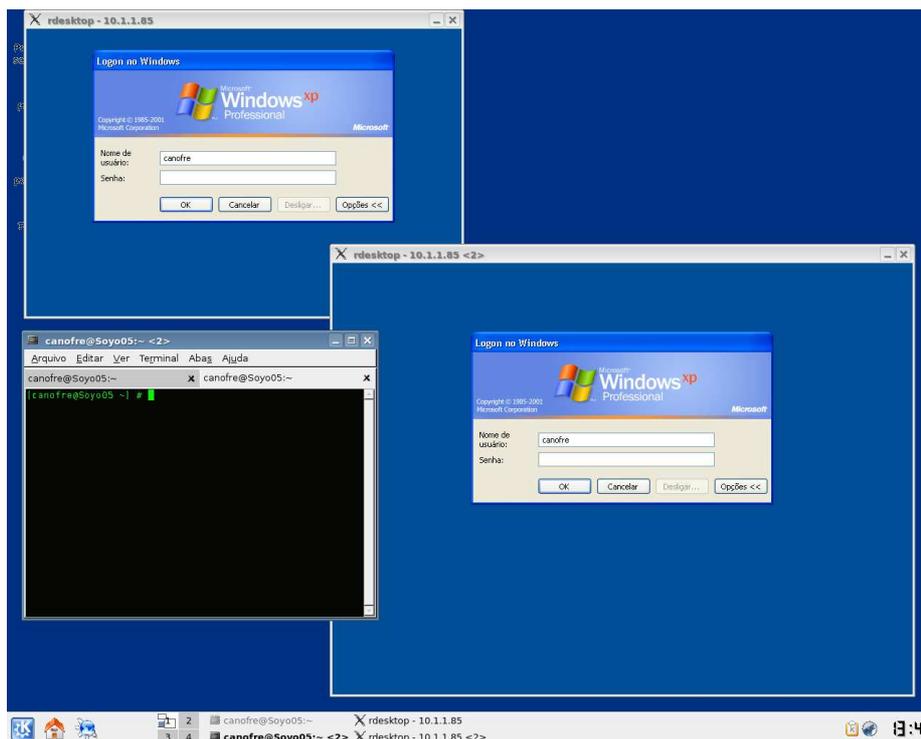


Figura 4.6: Execução do SO Windows em uma janela com resolução 800x600.

Mycosystems disponibilizar uma ferramenta que possibilita a execução desse ambiente nos *Thin Clients*, o SCW, a sua disponibilidade para os usuários do NCC tornou-se um item desejável.

Para a execução do SO Windows nos terminais, foi necessária a instalação de algumas atualizações disponibilizadas pela SM para o Solaris e a instalação e configuração do *Sun Ray Connector for Windows* nos servidores, de acordo com o manual de instalação e configuração do produto (Sun Microsystems, 2007e). Desta forma, em uma DTU executando SO Solaris passou a ser possível executar o SO Windows. Para disponibilizar o mesmo funcionamento para um *Thin Client* executando SO Linux, foi ainda necessário instalar o pacote *rdesktop* nas máquinas anfitriãs.

A utilização do SO Windows através dos terminais, independente do SO utilizado, e realizada através da linha de comando, onde são especificados argumentos que permitem executar o sistema em uma janela como pode ser visto na figura 4.6 ou em tela cheia. Dessa forma, para facilitar o uso dos comandos pelos usuários, foi criado um *script* que por padrão abre uma janela com resolução de 800 X 600 *pixels*, podendo ser utilizada uma outra resolução, o que permite ao usuário utilizar a transferência de conteúdo de texto entre o Windows e o SO sobre o qual ele esta sendo executado.

4.4 O Sun Ray Desktop Manager

Visando facilitar o gerenciamento da rede de terminais, incluindo além dos servidores e terminais, as máquinas anfitriãs, era parte do objetivo deste trabalho a utilização do SDM, uma interface administrativa disponibilizada pela Sun, que visa gerenciar tanto as DTU's quanto suas máquinas anfitriãs.

A utilização desta ferramenta segundo seus manuais (Sun Microsystems, 2007e), permitiria gerenciar perfis de usuários e grupos de usuários através da utilização agentes e ferramentas de gestão de rede, com o auxílio de uma base LDAP. Ela também proporcionaria a administração de exportação e compartilhamento de arquivos através do sistema de arquivos ZFS (*Zettabyte File System*)⁵, dentre a administração de outras funcionalidade do servidor Solaris no qual estiver instalada.

Dentre as aplicações suportadas para a criação e manipulação desses perfis encontram-se: Gconf (*Gnome configuration framework*), StarOffice⁶ e Mozilla Preferences. Podendo ainda ampliar as possibilidades de gerencia dos perfis, através da utilização de modelos, disponibilizados com a instalação de extensões do *software*.

A utilização do SDM era composta de duas partes, primeiramente a instalação e configuração do servidor e posteriormente a do clientes mas máquinas anfitriãs. No entanto a realização da segunda parte não pode ser efetivada, devido a ferramenta em questão não possuir uma versão para SO Linux, impedindo desta forma sua configuração e análise.

No entanto, buscando verificar a possibilidade de se obter alguma das funcionalidades mencionadas no manual do SDM, foi instalada a parte referente ao servidor que conforme pode ser vista na figura 4.7, sem os clientes instalados e configurados, disponibiliza somente o gerenciamento de arquivos ZFS, o qual também fez parte das tentativas de exportar os diretórios montados pelos *pendrive*, não obtendo sucesso.

Outra dificuldade verificada através desta ferramenta, foi referente a utilização do servidor LDAP utilizado pelo curso de Ciência da Computação, cujos dados utilizados para sua configuração (endereço e porta), não foram aceitos pelo SCW. No entanto esta dificuldade não foi analisada de forma mais aprofundada devido ao serviço não oferecer possibilidades de uso sem os clientes configurados. Esses fatores impossibilitaram a utilização desta ferramenta até mesmo para uma análise de suas possíveis funcionalidades.

⁵Sistema de arquivos da *Sun Microsystems* para o SO Solaris

⁶Pacote office da *Sun Microsystems*



Figura 4.7: Interface gráfica administrativa do SDM.

4.5 Quadro resumo do trabalho desenvolvido

A tabela 4.3 apresenta um resumo do ambiente desenvolvido, informando os objetivos propostos no início deste trabalho o seu estado atual e algumas observações caso necessário.

Objetivo Proposto	Estado atual	Observações
Uso do SRSS no SO Solaris para gerenciamento dos terminais	Em funcionamento	Os terminais já são gerenciados pelo SRSS
Uso da Interface administrativa <i>web</i> do SRSS	Em funcionamento	
Acesso a dispositivos de armazenamento em massa	Em funcionamento	Com a utilização de <i>script</i>
Uso dos periféricos de áudio	Não disponível	Para o acesso a esse dispositivo é necessária a utilização dos seus respectivos <i>drivers</i> , o que não foi buscado devido a solução de outros itens terem uma maior importância
Configuração de um grupo <i>failover</i>	Em funcionamento	Com a utilização de dois servidores
Acesso ao SO Windows	Em funcionamento	Com o auxílio do <i>rdesktop</i>
Utilização do SDM	Não disponível	Não foi possível a utilização desta ferramenta devido a inexistência de <i>software</i> cliente para o SO Linux

Tabela 4.3: Quadro resumos do trabalho desenvolvido.

5 CONCLUSÃO

Com a disponibilização de computadores com maior poder computacional e a um menor custo, o acesso e a utilização dos mesmos tem se tornado muito freqüente. Dessa forma, muitas máquinas acabam por vezes tendo seu poder computacional sub utilizado, devido a simplicidade das tarefas para as quais são destinadas. Mesmo com um custo mais acessível, a renovação ou ampliação de um ambiente profissional ou educacional, continua exigindo um relativo investimento, que na maioria das vezes não se dispõe ou não se deseja dispor.

Com a utilização de recursos computacionais virtualizados, é possível obter uma melhor utilização de *hardware* e *software* existente em uma máquina real, mostrando-se assim como uma alternativa ao melhor aproveitamento das máquinas existentes. Visto que várias máquinas podem executar simultaneamente sobre um único computador real, diminuindo assim os investimentos necessários.

Unindo-se poder computacional e virtualização, com a tecnologia de *Thin Client*, torna-se possível prover as necessidades de instituições de uma forma em geral, satisfazendo as suas possibilidades. Pois com um baixo investimento pode-se facilmente dobrar o número de máquinas em um laboratório ou ambiente qualquer e mantê-lo com um investimento em manutenção consideravelmente menor do que o necessário para se manter um ambiente com PC's completos.

No entanto, o uso destes terminais atrelados a virtualização, não proporciona somente vantagens para as instituições, mas também para os administradores de rede, pois facilita o gerenciamento das estações de trabalho, concentrando em um servidor ou em algumas máquinas o trabalho pesado.

Uma das contribuições deste trabalho foi a disponibilização da utilização dos *Thin Clients*, proporcionando ainda uma certa gerenciabilidade para a inclusão de novos terminais

ou máquinas anfitriãs. A disponibilidade dos terminais também foi garantida através da implementação de um grupo de servidores tolerantes a falhas, ou seja, o recurso *failover group* oferecido pela ferramenta utilizada, impedindo assim que os terminais entrassem em um estado indisponível, no caso de falha do servidor. Também com fator importante este trabalho disponibilizou o acesso a dispositivos usb, o qual encontra-se configurado permitindo o acesso a dispositivos como *pendrive*.

Outra contribuição importante deste trabalho foi proporcionar a utilização dos SO Windows através das DTU's, possibilitando o uso deste sistema em uma mesma máquina utilizada para executar o SO Linux, não se fazendo necessário a instalação de dois sistemas em uma mesma máquina ou de uma divisão de máquinas. Isso foi possibilitado através do *Sun Ray Connector for Windows* que também permite que os dois sistemas sejam utilizados simultaneamente em um *Thin Client*.

Com pontos negativos deste trabalhos são ressaltadas as dificuldades encontradas ao se trabalhar com o SO Solaris que apresenta algumas peculiaridades com relação ao comandos e configurações, com relação aos SO mais comuns, que dificultou a apresentação de melhores resultados. E também a incompatibilidade do *Sun Ray Desktop Manager* com o SO Linux e o servidor de autenticação da administração do NCC, impossibilitando a sua utilização.

Comparando-se as disponibilidades e configurações do ambiente anteriormente configurado, com o ambiente atual, pode-se observar que o objetivo do trabalho foi alcançado em sua maioria. Excluindo-se a dificuldade inicial da utilização de um sistema Unix, o conjunto de ferramentas disponibilizadas para gerenciamento dos terminais, é capaz de proporcionar facilidades na gerência de sua rede, proporcionando também uma melhor utilização dos recursos destes dispositivos.

REFERÊNCIAS

BARHAM, P.; DRAGOVIC, B.; FRASER, K.; HAND, S.; HARRIS, T.; HO, A.; NEUGEBAUER, R.; PRATT, I.; WARFIELD, A. Xen and the art of virtualization. **ACM Symposium on Operating Systems Principles**, [S.l.], v.19, p.164–177, 2003.

GOLDBERG, R. P. Survey of virtual machine research. **IEEE Computer Magazine**, [S.l.], v.7, n.6, p.34–35, june 1974.

GOOGLE. **Produtos e Ferramentas do GOOGLE**. Disponível em: <<http://www.google.com/options/>>. Acesso em: 22 novembro 2007.

KOSLOVSKI, G. P. **DESENVOLVIMENTO DE UMA FERRAMENTA PARA ADMINISTRAÇÃO DE ARQUITETURAS VIRTUALIZADAS BASEADAS EM XEN**. Santa Maria: Monografia (Trabalho de Graduação) — Curso de Ciência da Computação, Universidade Federal de Santa Maria, 2007.

KOSLOVSKI, G. P.; BOUFLEUR, M. P.; CHARÃO, A. S. Uso de Virtualização de Recursos Computacionais na Administração de Redes. **Anais da IV Escola Regional de Redes de Computadores**, Passo Fundo, v.4, 2006.

Microsoft Corporation. **Microsoft Virtual PC**. Disponível em: <<http://www.microsoft.com/windows/products/winfamily/virtualpc/overview.mspx/>>. Acesso em: 22 novembro 2007.

NOGUEIRA, M. **Thin Client - Tecnologia**. Disponível em: <<http://marcelonogueira.wordpress.com/category/tecnologia/>>. Acesso em: 25 novembro 2007.

OPENIT. **FreeBSD Handbook**. Disponível em: <<http://www.openit.com.br/freebsd-hb/>>. Acesso em: 25 novembro 2007.

OpenVZ. **OpenVZ**. Disponível em: <<http://openvz.org/>>. Acesso em: 22 novembro 2007.

PC&CIA. **Computação Leve**: reduza o preço da informática no seu orçamento. Disponível em: <<http://www.wsystem.com.br/downloads/diversos/PC&Cia%20Junho%202007.pdf/>>. Acesso em: 25 novembro 2007.

SAADE, J. **Bash Guia de Consulta Rápida**. São Paulo: Novatec Editora Ltda., 2001.

SANTOS, R. C. M. dos; SCHIMIDT, A. H.; BOUFLEUS, M. P.; CHARÃO, A. S. Análise de Desempenho da Virtualização de Rede nos Sistemas Xen e OpenVZ. **Anais da V Escola Regional de Redes de Computadores**, Santa Maria, v.5, 2007.

SOUZA FILHO, J. de. **ANÁLISE COMPARATIVA ENTRE AMBIENTES COMPUTACIONAIS BASEADOS NAS TECNOLOGIAS DE PROCESSAMENTO DISTRIBUÍDO E DE THIN CLIENT**. Blumenau: Monografia (Trabalho de Conclusão de Curso) — Curso de Engenharia de Telecomunicações, Universidade Regional de Blumenau, 2003.

Stanford University. **SimOS - The Complete Machine Simulator**. Disponível em: <<http://simos.stanford.edu/>>. Acesso em: 22 novembro 2007.

Sun Microsystems. **Sun Microsystems**. Disponível em: <<http://www.sun.com/>>. Acesso em: 01 novembro 2007.

Sun Microsystems. **Guia de Instalação e Administração do Sun Ray Connector for Windows Operating Systems 2.0**. [S.l.: s.n.], 2007. Disponível em: <http://docs.sun.com/app/docs/doc/820-2031?l=pt_BR>. Acesso em: 16 novembro 2007.

Sun Microsystems. **Sun Desktop Manager 1.0 Installation Guide**. [S.l.: s.n.], 2007. Disponível em: <http://docs.sun.com/app/docs/doc/819-2725?l=pt_BR>. Acesso em: 16 novembro 2007.

Sun Microsystems. **Guia de Instalação e Configuração do Sun Ray Server Software 4.0 para o Sistema Operacional Solaris**. [S.l.: s.n.], 2007. Disponível em: <http://docs.sun.com/app/docs/doc/820-1995?l=pt_BR>. Acesso em: 16 Janeiro 2007.

Sun Microsystems. **Sun Ray Server Software 4.0 Administrator's Guide for the Solaris Operating System**. [S.l.: s.n.], 2007. Disponível em: <http://docs.sun.com/app/docs/doc/820-0411?l=pt_BR>. Acesso em: 20 Janeiro 2007.

Sun Microsystems. **Sun Desktop Manager 1.0 Administration Guide**. Disponível em: <http://docs.sun.com/app/docs/doc/819-2726?l=pt_BR>. Acesso em: 2007.

Sun Microsystems. **Sun Ray Quick Reference**. Disponível em: <<http://dlc.sun.com/pdf/805-7871-12/805-7871-12.pdf>>. Acesso em: 15 janeiro 2008.

Sun Ray Community. **Sun Ray Community**. Disponível em: <http://wiki.sun-rays.org/index.php/Sun_Ray_Community>. Acesso em: 20 janeiro 2008.

TORRES, G. **Redes de Computadores Curso Completo**. Rio de Janeiro: Axcel Books, 2001.

VMware. **VMware**. Disponível em: <<http://www.vmware.com/>>. Acesso em: 12 novembro 2007.

WATERS, J. K. **ABC da Virtualização**. Disponível em: <<http://cio.uol.com.br/tecnologia/2007/08/14/idgnoticia.2007-08-14.5515750576/>>. Acesso em: 19 agosto 2007.

WHATELY, L. L. A.; AMORIM, C. L. Sistemas de Computação Baseados em Máquinas Virtuais. **International Symposium on Computer Architecture and High Performance Computing**, [S.l.], v.17, p.34–35, 2005.

WIKIPÉDIA. **VMware**. Disponível em: <<http://pt.wikipedia.org/wiki/VMware/>>. Acesso em: 23 novembro 2007.

WIKIPÉDIA. **Bare Metal**. Disponível em: <<http://en.wikipedia.org/>>. Acesso em: 24 novembro 2007.

WIKIPÉDIA. **Thin Client**. Disponível em: <http://pt.wikipedia.org/wiki/Thin_client>. Acesso em: 25 novembro 2007.

YOUSEFF, L.; WOLSKI, R.; GORDA, B.; KRINTZ, C. **Paravirtualization for HPC Systems**. [S.l.]: UCRL-TR-225347, Lawrence Livermore National Laboratory (LLNL), Livermore, CA, 2006.