Centro Universitário de Brasília – UniCEUB
Faculdade de Ciências Exatas e Tecnologia – FAET

Projeto Final de Graduação do

Curso de Engenharia da Computação

Título:

Gerenciamento de Falhas de Hardware em um Dispositivo de Rede utilizando SNMP

Autor:

Haldane Capanema Abreu

RA: 9965954

Orientador:

Prof. M.Sc. Antonio José Gonçalves

Brasília – DF

Junho/2005

Agradecimentos

Na conclusão de mais uma etapa da minha vida, tenho o maior prazer em agradecer:

- ✓ Primeiramente, a Deus, que é o meu criador, a minha força e proteção.
- ✓ A Jesus Cristo, que é o caminho, a verdade e a vida.
- ✓ Aos meus pais, que através do amor, dedicação e esforço, ajudaram-me a enfrentar e vencer os obstáculos da vida.
- ✓ À minha esposa, Sara, pelo amor, incentivo, compreensão e ajuda em todos os momentos.
- ✓ À minha mãe, Vera, por ser uma mulher forte, educada e carinhosa.
- ✓ Ao meu pai, Márcio, por ser um homem trabalhador e honesto.
- ✓ À minha irmã, Kelley, por sempre contribuir para o meu sucesso e, também, por sempre colaborar para a solução dos problemas.
- ✓ Ao meu irmão, Hamistause, pela ajuda prestada nas horas necessárias.
- ✓ Aos colegas de trabalho, Cleone, Wanderley e Roberto, pelos conhecimentos compartilhados em prol do meu projeto.
- ✓ E a todos os amigos, que de alguma forma contribuíram para a conclusão deste projeto.

Resumo

Este trabalho apresenta os estudos, as especificações e a arquitetura implementada no desenvolvimento de uma ferramenta de gerenciamento de falhas de hardware em um dispositivo de rede.

Para a coleta das informações dos objetos gerenciados, o SNMP foi adotado como protocolo de gerenciamento e, também, foram utilizados os objetos da MIB-II.

Um aspecto importante da metodologia foi a opção pela utilização de Softwares Livres, tais como, o sistema operacional Linux, o servidor Web Apache, a linguagem de desenvolvimento PHP, o banco de dados MySQL e o pacote Net-SNMP.

Como resultado do trabalho, foi implementado um protótipo para auxiliar o gerenciamento de falhas em uma rede de computadores.

Palavras-chave: SNMP, MIB, gerenciamento, falhas, redes.

Abstract

This work presents the studies, the specifications and the architecture implemented in the development of a fault management tool of an hardware in a net device.

For the collection of the information of managed objects, the SNMP was adopted as management protocol and, also, the objects of the MIB-II had been used.

An important aspect of the methodology was the option for the use of Free Software, such as, the operational system Linux, the Apache Web server, the development of PHP language, the MySQL data base and the package Net-SNMP.

As results, an archetype was implemented to assist the fault management in a network computer.

Word-key: SNMP, MIB, management, fault, network.

Índice

Lista de figuras e tabelas

Lista de abreviaturas

IEEE – *Institute of Electrical and Electronics Engineers*

IETF – *Internet Engineering Task Force*

IP – *Internet Protocol*

ISO – *International Organization for Standardization*

MIB – *Management Information Base*

OSI – *Open Systems Interconnection*

PDU – *Protocol Data Unit*

RFC – *Request For Comments*

SNMP – *Simple Network Management Protocol*

TCP – *Transmission Control Protocol*

UDP – *User Datagram Protocol*

WWW – *World Wide Web*

1 – Introdução

A utilização de redes de computadores pela sociedade tem crescido de forma exponencial nos últimos anos, e o uso de suas aplicações tem sido imprescindível na realização de inúmeras tarefas em vários ambientes de trabalho. A utilização harmônica das tecnologias de telecomunicações e informática, de natureza e porte diferentes, vem atingindo atualmente um estágio de grande amadurecimento.

Entretanto, com a utilização dos recursos de rede existentes e a inclusão de inúmeras tecnologias e novas aplicações, a degradação da performance da rede cresceu. Em muitos casos, porém, esta degradação deveu-se exclusivamente ao mau gerenciamento dos recursos disponíveis.

Atualmente as redes de computadores e os seus recursos associados, além das aplicações distribuídas, têm se tornado um recurso fundamental. Tal é a sua importância para uma organização, que elas basicamente não podem falhar.

Isto significa que o nível de falhas e de degradação de desempenho considerados aceitáveis está cada vez menor, podendo este nível tender a zero, dependendo da importância da rede para a instituição.[Ammirabile, 2004]

A área de gerência de redes foi impulsionada pela necessidade de monitoração e controle do universo de dispositivos que compõem as redes de comunicação. Por sua vez, a gerência de falhas, cuja finalidade é localizar, determinar as causas e corrigir as falhas na rede de computadores, é uma das funções chave dos sistemas de gerenciamento. [Salgues, 2004]

Atualmente, uma das preocupações da área de gerência de redes é o desenvolvimento de recursos e ferramentas voltadas para as atividades de gerenciamento. É esse tipo de preocupação que motivou o desenvolvimento de uma ferramenta de gerenciamento de falhas de hardware em um dispositivo de rede.

A objetivo deste projeto final de curso consiste em adquirir conhecimentos teóricos e experiência prática sobre gerenciamento de redes de computadores, abrangendo desde os estudos sobre gerenciamento, SNMP, MIBs, passando pelas ferramentas e utilitários Linux, Apache, Net-SNMP, PHP e MySQL, indo até a criação de um protótipo de

gerenciamento. Além disso, pretende-se mostrar, através de uma simulação de gerenciamento, a comunicação entre um Gerente e um Agente utilizando o protocolo SNMP.

Espera-se que este trabalho possa incentivar outros trabalhos e que no futuro, auxilie o autor a gerar idéias e soluções no ambiente profissional.

2 – Gerenciamento de Redes

2.1 – O que é Gerenciamento de Redes

A área de gerenciamento de redes está se expandindo, incluindo o gerenciamento de outros serviços, não considerados como parte da gerência de redes. Dentre esses serviços destacam-se:

- ✓ Gerenciamento de Sistemas: computadores pessoais e seus componentes;
- ✓ Gerenciamento de Serviços: serviços de rede e seus equipamentos de gerência;
- ✓ Gerenciamento de Aplicações: aplicações desenvolvidas e utilizadas pelos usuários;
- ✓ Gerenciamento de Recursos: banco de dados, dispositivos de armazenamento, correio eletrônico, dentre outros.

As atividades de gerenciamento de redes consistem na detecção e correção de erros em um tempo mínimo e no estabelecimento de procedimentos para a previsão de problemas futuros. A eficiência na realização de todas as atividades de gerência está diretamente ligada à presença de ferramentas que as automatizem e de pessoal qualificado. [Ammirabile, 2004]

A primeira geração de plataformas de gerenciamento de redes tinha o objetivo de monitorar os principais dispositivos da rede e um conjunto de estatísticas de operações. As ferramentas para controlar esses dispositivos eram limitadas, principalmente devido à falta de segurança e as limitações gerais de como a ferramenta havia sido implementada.

Hoje, as consoles de gerência usam interfaces gráficas para apresentar visões diferentes da rede. Ferramentas de gerência e aplicações selecionam informações úteis de vários dados de gerência. Os resultados são armazenados em banco de dados para posterior análise. Muitas aplicações utilizam técnicas de *software* avançadas para tornar a informação mais clara e concisa. A interpretação da informação de gerenciamento permite a análise de problemas da rede mais diretamente.

2.2 – Elementos do Modelo de Gerência

No ambiente de gerência de redes existem três conceitos importantes: gerentes, agentes e objetos gerenciados. Um gerente pode obter informações atualizadas sobre os objetos gerenciados e controlá-los. Para isso, transmite operações de gerenciamento aos agentes. Um agente executa operações de gerenciamento sobre objetos gerenciados. Pode, ainda, transmitir ao gerente as notificações emitidas pelos objetos gerenciados. No entanto, estes gerentes e agentes não possuem nenhuma autonomia, ou seja, não participam do processo de tomada de decisão. A tomada de decisão é normalmente realizada por um especialista (administrador da rede) [Ammirabile, 2004; Salgues, 2004]. A figura 2.1 mostra os componentes do modelo de gerência.

Figura 2.1 – Componentes do modelo de gerência. [Ammirabile, 2004]

2.2.1 – Gerente

O gerente compreende um tipo de software que permite a obtenção e o envio de informações de gerenciamento, junto aos mecanismos gerenciados, mediante comunicação com um ou mais agentes. As informações de gerenciamento podem ser obtidas com o uso de requisições efetuadas pelo gerente ao agente, como também, mediante envio automático disparado pelo agente a um determinado gerente (mensagens denominadas *Traps*). Tipicamente um gerente está presente em uma estação de gerenciamento de rede.

2.2.2 – Agente

O agente compreende um tipo de software presente junto aos dispositivos gerenciados. A função principal de um agente baseia-se no atendimento das requisições enviadas por um software gerente e o envio automático de informações de gerenciamento ao gerente, indicando a ocorrência de um evento previamente programado. Também compete ao agente efetuar a interface entre os diferentes mecanismos usados na instrumentação das funcionalidades de gerenciamento inseridas em um determinado dispositivo gerenciado.

2.2.3 – SNMP

O SNMP - *Simple Network Management Protocol* é um protocolo da camada de aplicação designado para facilitar a troca de informações de gerenciamento entre dispositivos de rede. Usando os dados transportados pelo SNMP, os administradores de rede podem gerenciar mais facilmente o desempenho da rede, encontrar e solucionar problemas e planejar com mais precisão uma possível expansão da mesma [Ammirabile, 2004].

O protocolo de gerenciamento, SNMP, compreende o conjunto de regras e formatos de mensagens. Os mecanismos de comunicação entre gerentes e agentes são implementados com base nas especificações deste protocolo.

Atualmente, o SNMP é o mais popular protocolo para gerenciamento de diversas redes. Esta popularização se deve ao fato de que o SNMP é um protocolo relativamente simples, porém suficientemente poderoso para resolver os difíceis problemas apresentados quando se gerencia redes heterogêneas.

A primeira versão de SNMP foi formulada em 1988, e oficialmente publicada em 1990, na forma de três normas técnicas – RFCs [17]. Ele logo se tornou um padrão de fato para a inclusão de "inteligência", em forma de MIBs e agentes, em produtos comerciais para a interconexão de redes, especialmente roteadores, pontes e concentradores. Até 1992 haviam sido definidas outras duas dúzias de normas para as MIBs específicas de diferentes tipos de

equipamentos, interfaces de telecomunicações e protocolos, os quais poderiam ser gerenciados utilizando o SNMP. Por ser a primeira solução disponível, o SNMP, embora uma solução simples, foi um enorme sucesso porque claramente preencheu um vazio. O esquema CMIS/CMIP da abordagem OSI, definida mais ou menos na mesma época e com funcionalidade mais extensa do que o SNMP, não teve tão ampla adoção para o gerenciamento das grandes redes, provavelmente pelo aparecimento tardio de suas implementações, especialmente em forma de agentes em equipamentos de conectividade.

A simplicidade do SNMP, que o ajudou a ocupar o campo de gerenciamento das grandes redes, é a principal responsável pelas suas limitações. A segunda versão do SNMP, conhecida como SNMPv2 e lançada em 1993, consertou várias deficiências da versão original, nos campos de eficiência, segurança, flexibilidade e na distribuição da sua administração. Com estas melhorias, o SNMPv2 se tornou uma ferramenta afinada com as necessidades das redes grandes dos dias de hoje. Entre os objetivos de SNMPv2 está também a coexistência com a versão original, e sua utilização em ambientes diferentes de TCP/IP, inclusive em redes OSI, AppleTalk e NetWare [Ammirabile, 2004].

Em uma rede gerenciada segundo a arquitetura SNMP, podem-se distinguir os seguintes elementos específicos de gerência de rede:

- ✓ Vários equipamentos, dispositivos e softwares que contêm, cada um, uma entidade de processamento SNMP chamada de agente;

- ✓ Pelo menos uma estação gerenciadora (gerente) que contém a inteligência e o domínio das operações de gerência. Esta estação executa aplicações de gerência que monitoram e controlam os elementos da rede, por meio de acesso remoto à informação de gerenciamento armazenada nesses elementos;

- ✓ Um protocolo de comunicação, SNMP, utilizado para transferir informações de gerenciamento entre agentes e gerentes.

Os agentes possuem um repositório de informações de gerência que é chamado MIB (*Management Information Base* - Base de Informações de Gerenciamento). É de responsabilidade do agente manter na MIB um reflexo da realidade operacional e administrativa do dispositivo gerenciado. Desta forma, o gerente pode saber o que está acontecendo por meio de pedidos de informações da MIB do agente. A estratégia implícita do

SNMP é o *polling* das informações nos diversos agentes, ou seja, o gerente faz constantemente o pedido de informações sobre os elementos gerenciados.

A organização das informações na MIB de um agente SNMP é estruturada na forma de objetos gerenciados. Estes objetos, que são unidades de armazenamento de dados, representados na MIB por suas instâncias únicas ou múltiplas, compõem a base de informações de gerência referente àquele determinado elemento da rede.

Assim, o modelo arquitetural SNMP é centralizado e composto de estações de gerência e elementos de rede, estes contendo ou sendo representados por agentes SNMP. O agente pode estar localizado nos próprios elementos de rede, como roteadores e servidores; ou pode ser colocado em algum componente que monitora os elementos de rede, a exemplo da situação onde um hub é monitorado por um agente residente num servidor ligado a este hub.

Vale observar que cabe à estação gerente fazer a coleta de dados de cada elemento gerenciado por intermédio do agente correspondente. O protocolo SNMP é então responsável pelo transporte das informações de gerenciamento entre o gerente e os agentes existentes nos elementos de rede [Comer, 1998].

2.2.3.1 – Elementos SNMP

O SNMP é uma parte da arquitetura de gerenciamento da Internet, baseada na interação de diversas entidades, como se segue [Ammirabile, 2004; Salgues, 2004]:

- ✓ Elementos de rede – também chamados dispositivos gerenciados, os elementos de rede são dispositivos de *hardware* como os computadores, roteadores, e servidores de terminais que estão conectados a rede.

- ✓ Agentes – são módulos de *software* que residem nos elementos de rede. Eles coletam e armazenam informações de gerenciamento como o número de pacotes de erros recebidos pelo elemento de rede. São eles que respondem às solicitações dos gerentes.

- ✓ Objeto gerenciado – um objeto gerenciado é qualquer elemento que possa ser gerenciado.

✓ MIB – uma MIB é uma coleção de objetos gerenciados residentes em um armazenamento virtual de informações. Coleções de objetos gerenciados relacionados são definidas em módulos específicos da MIB.

✓ Notação sintática – é a linguagem usada para descrever os objetos gerenciados da MIB em um formato independente da plataforma. Um uso consistente da notação sintática permite que diferentes tipos de computadores compartilhem informações. Sistemas de gerenciamento Internet usam um subconjunto da *Open System Interconnection* (OSI), o *Abstract Syntax Notation 1* (ASN.1) da *International Organization for Standardizations* (ISO), para definir tanto os pacotes que são trocados pelo protocolo de gerenciamento quanto os objetos que ele deve gerenciar.

✓ *Structure of Management Information* (SMI) – o SMI define as regras para descrever as informações de gerenciamento. O SMI é definido usando ASN.1.

✓ *Network Management Stations* (NMS) – também chamados de gerentes, estes dispositivos executam aplicações de gerenciamento para monitorar e controlar elementos de rede.

✓ Protocolo de gerenciamento – um protocolo de gerenciamento é usado para transportar informações de gerenciamento entre agentes e NMS. O SNMP é o protocolo de gerenciamento padrão da comunidade Internet.

2.2.3.2 – Operações SNMP

O SNMP é um protocolo de requisição e resposta simples. Algumas operações são definidas no SNMP:

✓ *Get* – permite que o gerente recupere uma instância de objeto em um agente.

✓ *Walk* – permite que o gerente recupere uma ou mais instâncias de objeto em um agente.

✓ *GetNext* – permite que o gerente recupere a próxima instância de objetos de uma tabela ou lista em um agente. Se o gerente quiser recuperar todos os elementos de uma tabela de um agente, ele inicia com uma operação *Get* seguida de uma série de operações *GetNext*.

- ✓ *Set* – permite que o gerente modifique valores de uma instância de objetos em um agente.

- ✓ *Trap* – usado pelo agente para informar o gerente, de modo assíncrono, sobre algum evento.

Os pacotes de mensagem do SNMP são divididos em duas partes. A primeira parte contém a versão e o nome da comunidade. A segunda parte contém a Unidade de Dados do Protocolo – PDU do SNMP especificando a operação que será realizada (*"Walk", "Get"*, "*Set*" e outros) e a instância de objetos envolvida na operação.

2.3 – MIB

O conceito de objeto gerenciado é muito importante em gerência de redes, sendo definido como a representação das características relativas à gerência de um recurso físico ou de um recurso lógico.

Dependendo da granularidade que se deseje na gerência dos recursos, um objeto gerenciado pode modelar um recurso real, como um roteador, por exemplo, ou uma relação entre recursos, como uma rota, entre outros. A definição de um objeto apresenta dois aspectos: onde ele se situa (sua localização dentro do sistema sendo gerenciado) e sua natureza representada por seus atributos [Kurose, 2003].

A MIB – *Management Information Base* é o repositório dos objetos gerenciados dentro de um sistema, não importando o meio para armazenamento físico das informações de gerência.

A MIB compreende um conjunto de variáveis usadas para representar informações estáticas ou dinâmicas vinculadas a um determinado dispositivo gerenciado. Grande parte das funcionalidades de um software gerente/agente destina-se à troca de dados existentes na base de informações de gerenciamento.

2.3.1 – MIB-II

A MIB-II, contida na RFC 1213, define a segunda versão da base de informação de gerenciamento – MIB.

A árvore hierárquica, figura 2.2, foi definida pela ISO e representa a estrutura lógica da MIB-II. No topo da hierarquia está a ISO(1). No ramo ISO encontra-se registros para os padrões emitidos por entidades reconhecidas pela ISO (1.3). Nesses padrões, está o Departamento de Defesa dos Estados Unidos (1.3.6). Sob esse ramo está Internet (1.3.6.1), management (1.3.6.1.2) e o que queremos MIB-II (1.3.6.1.2.1), [Kurose, 2003]. Os números entre parênteses indicam cada nó da árvore. Por exemplo, a seqüência ISO.ORG.DOD é o mesmo que 1.3.6.

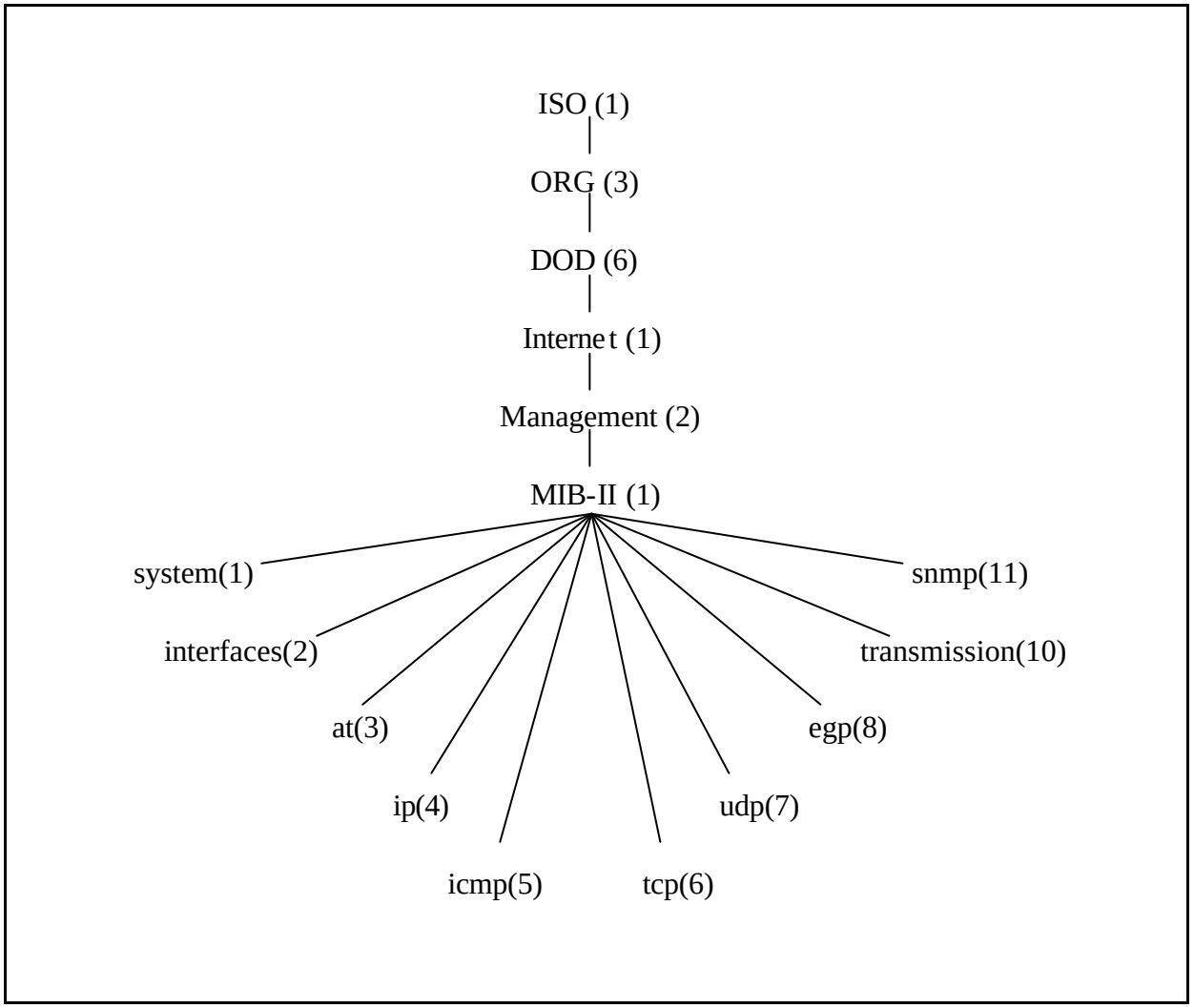

Figura 2.2 – Grupo MIB-II

O grupo MIB-II, conforme a figura 2.2, está subdividido em: system (1.3.6.1.2.1.1), interfaces (1.3.6.1.2.1.2), at (1.3.6.1.2.1.3), ip (1.3.6.1.2.1.4), icmp (1.3.6.1.2.1.5), tcp (1.3.6.1.2.1.6), udp (1.3.6.1.2.1.7), egp (1.3.6.1.2.1.8), transmission (1.3.6.1.2.1.10) e snmp (1.3.6.1.2.1.11).

A Tabela 2.1 mostra cada grupo da MIB-II e a sua respectiva descrição.

Tabela 2.1 – Grupos da MIB-II, RFC 1213

Grupo	**Descrição**
system (1)	Informações sobre o sistema
interfaces (2)	Informações sobre as interfaces
at (3)	Informações sobre a tradução de endereços
ip (4)	Informações sobre o protocolo IP
icmp (5)	Informações sobre o protocolo ICMP
tcp (6)	Informações sobre o protocolo TCP
udp (7)	Informações sobre o protocolo UDP
egp (8)	Informações sobre o protocolo EGP
transmission (10)	Informações sobre o meio de transmissão
snmp (11)	Informações sobre o protocolo SNMP

2.3.1.1 – Grupo System

O grupo System contém informação geral sobre o sistema gerenciado. Os objetos deste grupo são em grande parte explicativos [Stallings, 1999].

2.3.1.2 – Grupo Interfaces

O grupo Interfaces contém informação genérica sobre as interfaces físicas da entidade, inclusive informação de configuração e estatísticas nos eventos que acontecem a cada interface [Stallings, 1999].

2.3.1.3 – Grupo AT

O grupo AT (*Address Translation*) consiste no mapeamento do endereço de rede para um endereço físico. O endereço físico depende da natureza da subrede. Por exemplo, o endereço MAC e o endereço IP [Stallings, 1999].

2.3.1.4 – Grupo IP

O grupo IP (*Internet Protocol*) contém informação relevante para a implementação e operação do IP no ponto. Considerando que o IP é implementado em sistemas fim e sistemas intermediários, nem todos os objetos deste grupo são relevantes para determinado sistema. Objetos que não são relevantes possuem os valores nulos [Stallings, 1999].

2.3.1.5 – Grupo ICMP

O grupo ICMP (*Internet Control Message Protocol*) contém informação relevante para a implementação e operação do protocolo ICMP no ponto. O grupo contém contadores de vários tipos de mensagens ICMP enviadas e recebidas [Stallings, 1999].

2.3.1.6 – Grupo TCP

O grupo TCP (*Transmission Control Protocol*) contém informação relevante para a implementação e operação do protocolo TCP no ponto [Stallings, 1999].

2.3.1.7 – Grupo UDP

O grupo UDP (*User Datagram Protocol*) contém informação relevante para a implementação e operação do protocolo UDP no ponto. Há também informação sobre datagramas enviados e recebidos [Stallings, 1999].

2.3.1.8 – Grupo EGP

O grupo EGP (*External Gateway Protocol*) contém informação relevante para a implementação e operação do protocolo EGP no ponto. Há também informação sobre mensagens EGP enviadas e recebidas [Stallings, 1999].

2.3.1.9 – Grupo SNMP

O grupo SNMP (*Simple Network Management Protocol*) contém informação relevante para a implementação e operação do protocolo SNMP. Os objetos do SNMP possuem funções de gerente ou agente [Stallings, 1999].

2.4 – Áreas Funcionais de Gerenciamento

Em virtude da complexidade natural das redes de computadores, gerenciá-las de forma eficiente e eficaz representa um grande desafio. Preocupada em facilitar e organizar o desenvolvimento de projetos (construção e ou utilização de software) destinados ao gerenciamento de redes, a ISO propõe no Modelo de Gerenciamento OSI, a divisão das tarefas de gerenciamento em cinco áreas funcionais descritas a seguir [Stallings, 1999]:

- ✓ Gerenciamento de Falhas;
- ✓ Gerenciamento de Desempenho;
- ✓ Gerenciamento de Segurança;
- ✓ Gerenciamento de Contabilização;
- ✓ Gerenciamento de Configuração.

2.5 – Aplicações de gerenciamento

As aplicações de gerenciamento são programas que transformam os dados SNMP em informações usadas pelos usuários da aplicação gerente (administradores de rede, usuários dos serviços de rede, etc.). Essas aplicações compreendem uma variedade de programas que interagem com os agentes, via operações de gerenciamento, manipulam e formatam as mensagens de *trap* ou acessam o banco de dados. As aplicações de gerenciamento auxiliam no processamento e análise dos dados obtidos junto às aplicações agentes.

Estas aplicações de gerenciamento obtêm dados destinados a apoiarem os administradores e usuários dos serviços de rede na tomada de decisões inteligentes, fornecendo gerenciamento pró-ativo e estendendo as funcionalidades das plataformas de gerenciamento em muitas direções [Weber, 1997].

A primeira geração de aplicações de gerenciamento incluiu a geração de relatório, sistemas *trouble ticket* (cadastro de falhas) e alarmes, em casos de falhas nos pontos gerenciados. Uma segunda geração de aplicações de gerenciamento, muitas vezes, inclui sistemas especialistas que podem alertar o pessoal de rede quando sérios problemas ocorrem, antecipando possíveis falhas em várias partes da rede gerenciada e usando heurísticas para apoiar o planejamento, a operação, e fornecer dados valiosos que podem ser usados para o conjunto de informações que serão apresentadas na interface de usuário.

Um componente essencial das aplicações de gerenciamento é o uso de uma *Application Program Interface* (API) padronizada. Isso permite que aplicações de múltiplos fornecedores sejam incluídas em uma única plataforma de gerenciamento, visando oferecer um conjunto de aplicações para uma determina organização. A interface entre as aplicações de gerenciamento e as operações de gerenciamento deverá ser bem definida, aberta e baseada em padrões, sempre que possível [Ammirabile, 2004].

3 – Gerenciamento de Falhas

Gerência de Falhas (*FM – Fault Management*) detecta, localiza e corrige problemas no *hardware* ou *software* de rede. Determina e, normalmente registra, que uma falha ocorreu, determina sua localização e então tenta repará-la. Inclui também processos para relatar problemas a usuários finais e gerentes, assim como para controlar tendências relacionadas a problemas [Ammirabile, 2004].

Falhas são diferentes erros. Uma falha é uma condição anormal que requer atenção (ou ação) de reparar. Uma falha normalmente é indicada pelo fracasso de uma operação ou pela ocorrência de erros repetitivos. Certos erros podem acontecer ocasionalmente e não são considerados falhas. As falhas são catalogadas para análise posterior.

Implícito à determinação de que um dispositivo de rede esteja operando corretamente, está o conhecimento das características da falha de cada dispositivo. Cada dispositivo deve ter um indicador de falhas predefinido que pode ser dinamicamente alterados.

Rotinas de manutenção preventiva podem assegurar por longo tempo o funcionamento apropriado dos dispositivos de rede. Gerenciamentos de falhas sofisticados incluem a antecipação de fracassos. Isto pode ser tratado pela programação de diagnósticos rotineiros executados nos dispositivos. Dependendo da capacidade do dispositivo, estes testes podem variar desde o simples registro de que o dispositivo esteja operando a rigorosos conjuntos de testes de diagnóstico.

Entre as causas mais prováveis para falhas em uma rede estão: erros de projeto e implementação da rede, erros de sobrecarga, distúrbios externos, tempo de vida útil de equipamentos e má qualidade de alguns equipamentos.

A correção de falhas envolve métodos manuais e automáticos para resolver os problemas.

Existe uma variedade de ferramentas para atender aos requisitos de gerenciamento de falhas, inclusive ferramentas de monitoramento que alertam os gerentes

sobre os problemas, analisadores de protocolos para resolução de falhas e softwares para documentar e alertar os usuários sobre os problemas.

Resumindo, o gerenciamento de falhas tem como objetivo detectar e resolver rapidamente situações que degradam ou impedem o funcionamento da rede. Para isto, deve-se determinar a origem da falha e reparar ou substituir os componentes com falha.

Este tipo de gerenciamento de rede tem precedência sobre as demais áreas. Pode ser reativa, reage às falhas à medida que ocorrem, ou pró-ativa, buscando desenvolver ações antes que as falhas ocorram.

3.1 – Gerenciamento de falhas de hardware

O gerenciamento de falhas de hardware tem a principal função de monitorar o estado dos dispositivos físicos gerenciados. Além disso, visa atingir a confiabilidade e a disponibilidade necessárias para o bom funcionamento da rede.

A principal informação que este tipo de gerenciamento realiza é informar se o dispositivo gerenciado está UP (funcionando) ou DOWN (falhas). De posse desta informação o administrador sabe se a rede está funcionando normalmente, ou, se há alguma falha.

3.2 – Principais objetos usados para a gerência de falhas

Para o gerenciamento de falhas são utilizados os objetos gerenciados da MIB. Dentre estes, estão os objetos citados abaixo. Nota-se, que cada objeto é seguido por uma seqüência numérica, de acordo com a árvore hierárquica, conforme a seção 2.3.1.

- ✓ *sysUpTime* (1.3.6.1.2.1.1.3)

O *sysUpTime* informa há quanto tempo um sistema está funcionando. Através deste objeto, a aplicação de gerenciamento de falhas pode determinar quando houve a reinicialização. Com isso, identifica-se o motivo. Se foi por motivos operacionais ou se ocorreu alguma falha.

- ✓ *ifOperStatus* (1.3.6.1.2.1.2.2.1.8)

O *ifOperStatus* informa o estado operacional atual da interface (up / down). Através deste objeto, é possível identificar se as interfaces estão funcionando corretamente. Se estiver em *up*, tudo bem, caso contrário, em *down*, ocorreu alguma falha.

- ✓ *ifLastChange* (1.3.6.1.2.1.2.2.1.9)

O *ifLastChange* informa quando a interface entrou no seu estado operacional atual. Através deste objeto, é possível saber quando foi o último evento acontecido na interface e descobrir se houve falha.

- ✓ *ifInOctets* (1.3.6.1.2.1.2.2.1.10)

O *ifInOctets* informa o número total de octetos recebidos pela interface. Através deste objeto, é possível observar o tráfego de entrada da interface. De posse dessa informação, sabe-se os momentos em que podem ter ocorrido falhas na interface como, por exemplo, uma queda no link de comunicação.

- ✓ *ifOutOctets* (1.3.6.1.2.1.2.2.1.16)

O *ifOutOctets* informa o número total de octetos transmitidos pela interface. Através deste objeto, é possível observar o tráfego de saída da interface. A identificação de falhas pode ocorrer de forma análoga ao objeto *ifInOctets.*

3.3 – Detecção, diagnósticos e procedimentos para resolução de falhas de hardware

Algumas das principais falhas que podem ocorrer em uma rede e devem ser gerenciadas, podem acarretar vários sintomas dependendo da intensidade que ocorre a falha, da circunstância e da recursividade do acontecimento [Ammirabile, 2004].

Alguns problemas de redes podem ser prevenidos quando são utilizadas boas práticas de projeto, configuração e manutenção da rede, outros problemas não. Muitos problemas causam sinais e sintomas muito semelhantes, sendo impossível distinguir, antes de uma análise detalhada, qual o problema que está ocorrendo.

Normalmente, esses sintomas são detectados pelos usuários da rede, mas também é possível que um problema seja detectado na rede antes mesmo que qualquer usuário o perceba. Numa rede bem gerenciada, os problemas devem ser de conhecimento da equipe de redes antes que os usuários percebam, diminuindo ou até cessando o impacto desse problema na rede.

Quando se percebe que algum problema está ocorrendo ou poderá ocorrer, é preciso detectar o problema, diagnosticar e solucioná-lo o quanto antes. Infelizmente, nem os melhores sistemas de gerência de redes podem evitar todas as falhas e problemas em uma rede de computadores.

A seguir estão descritos os principais sintomas de problemas, os principais problemas que acarretam esses sintomas, como diagnosticá-los e os procedimentos para gerenciá-los de forma positiva. Será feita uma breve descrição de cada sintoma e de cada problema e, juntamente com o problema estarão os procedimentos para diagnosticá-los e gerenciá-los.

Sintoma é o comportamento anormal da rede, informando que algo estranho está ocorrendo, como conseqüência da existência de um problema. Os principais sintomas de falhas que podem ocorrem em uma rede estão descritos a seguir, seguidos dos problemas que podem acarretar cada sintoma.

3.3.1 – Interfaces indisponíveis

É possível que interfaces de equipamentos de interconexão de redes, como roteadores, não mais transmitam dados devido a defeitos físicos. No entanto, é possível também desativar uma interface, quando esta ficará administrativamente desabilitada. Isso quer dizer que o administrador da rede pode habilitar ou desabilitar uma interface de acordo com seus requisitos e/ou necessidades [Ammirabile, 2004].

A não transmissão de dados através da interface que se encontra desabilitada (administrativamente ou não) pode causar queda no desempenho da rede, indisponibilidade desse ponto de rede que está com a interface down e também tornar a rede menos confiável.

Em geral, o estado administrativo da interface é comparado com o seu estado operacional. Quando o estado administrativo não é igual ao estado operacional, algum problema nesta interface pode estar ocorrendo. Encontrar uma interface não operacional cujo estado administrativo indica que esta deveria estar ativa é indicação de falha na interface.

✓ Diagnóstico

Para verificar o estado operacional de uma interface pode-se utilizar o objeto *ifOperStatus*, como descrito na seção 3.2.

✓ Procedimentos

Não há como antecipar a indisponibilidade de uma interface de um equipamento em alguns casos. Esse problema pode ocorrer por intervenção humana, quando a interface é desativada administrativamente, por problemas no equipamento (hardware) ou por queda no circuito.

Não é possível prever quando o equipamento irá apresentar problemas de hardware. Isso depende da vida útil do equipamento, do ambiente, da forma como está instalado (temperatura, poeira, etc.) e, ainda, da manutenção deste. O que pode ser gerenciado é qual a recorrência do problema, isto é, a quantidade de vezes que a interface se encontra inativa por problema de hardware do equipamento e a troca das interfaces problemáticas, para resolver o problema.

A única forma de impedir que as interfaces sejam administrativamente desativadas indevidamente é restringir o acesso ao roteador (acesso lógico e físico) ao menor

número de pessoas possíveis. Sendo assim, a senha deve ser conhecida somente pelo administrador da rede para evitar acesso indevido ao equipamento.

Caso uma interface se encontre indisponível é preciso avaliar vários parâmetros antes de executar qualquer procedimento. Entre esses parâmetros estão:

- ✓ Verificar o cabo conectado à interface, no caso das interfaces Ethernet;
- ✓ Verificar o modem e o circuito, junto à operadora, no caso de interfaces Seriais;
- ✓ Verificar quando foi a última alteração no roteador (*ifLastChange*) e, se possível, detectar o que foi alterado, pois pode estar gerando a inatividade na interface;
- ✓ Verificar se a interface apresenta recorrência de indisponibilidade, uma vez que a causa desta pode ser um problema de hardware, sendo assim, necessária a manutenção ou troca da interface.

Outro procedimento que pode ser adotado é a redundância dos circuitos de rede e dos dispositivos de interconectividade, a fim de satisfazer-se os requisitos de disponibilidade. A redundância elimina qualquer possibilidade de haver um único ponto de falha, tendo como procedimento, duplicar componentes críticos e criar circuitos redundantes, evitando-se assim, a indisponibilidade da rede. A redundância é dispendiosa, além disso, ela acrescenta complexidade à topologia de rede, ao endereçamento e ao roteamento.

3.3.2 – Cabos rompidos ou danificados

Cabos de fibra óptica quebrados completamente não permitem a passagem de sinais de uma extremidade à outra, inibindo o funcionamento da rede. As microfraturas no cabo tornam a rede lenta, uma vez que causam uma grande quantidade de erros, o que é um dos principais sintomas de problemas no cabeamento [Ammirabile, 2004].

- ✓ **Diagnóstico**

O principal sinal de um cabeamento ruim ou com problema é a elevada taxa de erros. Então é necessário sempre observar e analisar a taxa de erros. Os cabos podem ser verificados com um testador de cabos.

- ✓ **Procedimentos**

Procedimentos que impedem fraturas e rompimentos de cabos de fibra ótica, assim como em qualquer tipo de cabeamento, consistem em um bom projeto físico de redes e na adequação e manutenção do local da passagem dos cabos.

As fraturas em cabos de fibra ótica podem ser reparadas utilizando-se técnicas de fiber splice. Essa técnica consiste na junção através de fusão ou utilizando um acoplador ótico dos dois lados na quebra do cabo, sendo que a fusão é uma técnica que resulta em uma menor perda.

3.3.3 – Conectores defeituosos

Problemas com conectores podem causar mau contato com o equipamento ao qual o cabo está conectado, levando à intermitência da rede. Em outras situações, a conseqüência pode ser a falta de conectividade ou a rede ficar lenta [Ammirabile, 2004].

- ✓ **Diagnóstico**

Assim como em problemas no cabeamento, o principal sinal de problemas com conectores é a elevada taxa de erros,

- ✓ **Procedimentos**

Assim como o cabeamento, um bom projeto físico de redes e a adequação e manutenção do local onde se encontram os equipamentos, fazem com que os conectores tenham vida útil prolongada, evitando os defeitos. Caso não tenha como impedir esse problema, é necessário a instalação de um novo conector, seguindo os padrões da norma de rede física.

3.3.4 – Equipamentos defeituosos

Equipamentos de interconexão de redes, deixando de realizar suas tarefas, podem apresentar comportamentos anormais ou total inoperância devido a uma falha de hardware.

Muitas vezes o equipamento aparenta estar com defeito, mas uma reinicialização restabelece sua operação normal. Estas instabilidades podem ser causadas por quedas rápidas de energia ou erros do próprio sistema operacional do equipamento. Outras vezes, porém, é preciso a reposição dos elementos defeituosos, ou a troca do próprio equipamento por um equipamento novo [Ammirabile, 2004].

- ✓ **Diagnóstico**

Um equipamento de interconexão de redes com defeito de hardware pode causar lentidão na rede, indisponibilidade, queda no desempenho, descartes dos pacotes e aumento da taxa de erros. Alguns sinais para se detectar o problema de hardware são:

- ✓ Equipamentos não operacionais: infelizmente esse sinal não pode ser tratado como um sinal diferencial. Um equipamento é considerado não operacional se a comunicação com ele não for possível. Portanto, a causa de um equipamento não estar operacional pode ser realmente defeito no equipamento, mas também pode ser outra causa que não envolva o equipamento, como um cabo rompido, por exemplo;

- ✓ Interfaces apresentando estado não operacional: quando o estado administrativo de uma interface está configurado para que ela esteja operacional, mas ela não funciona, certamente há algum problema. Em especial, quando um grupo de interfaces falha, o problema pode ser no equipamento de interconexão. Para se ter uma idéia da saúde do equipamento, é recomendada medir a utilização de seus recursos. Limiares de utilização desses recursos sendo excedidos são indicativos de falha no equipamento;

- ✓ Taxas elevadas de utilização de CPU e memória: se a utilização da CPU e da memória tiver muito superior da utilização habitual, é sinal de que algum problema pode estar ocorrendo no equipamento;

- ✓ Aumento do tráfego de broadcast/multicast: este tipo de tráfego pode ser gerado por um equipamento de interconexão defeituoso. O efeito do alto tráfego de broadcast/multicast é a saturação dos processadores dos roteadores.

✓ Procedimentos

Para que se possa determinar se há um problema no hardware do equipamento, é preciso avaliar a vida útil desse equipamento, as condições em que se encontra instalado (temperatura, localização no hack, poeira, etc.), se há acesso físico a esse equipamento por pessoas não autorizadas e, se existe um histórico problemático desses equipamentos.

A melhor forma de detectar um problema no hardware do equipamento é monitorar os alarmes de "up" e "down" gerados por ferramentas e também, monitorar a variável da MIB-II, *sysUpTime*, que indica quanto tempo o equipamento está operacional. Se existir uma recorrência de quedas e "reboots" diários, e não for detectado nenhum outro problema na rede, é bem provável que o hardware esteja apresentando alguma falha e precise ser reparado ou substituído.

4 – Desenvolvimento da ferramenta de gerenciamento

Este trabalho visou desenvolver uma ferramenta que pudesse ser capaz de gerenciar falhas de hardware em um dispositivo de rede. Como resultado, foi implementado um protótipo.

O protótipo final satisfaz os objetivos propostos no início do projeto para o gerenciamento de falhas de hardware em um dispositivo de rede.

A implementação deste protótipo foi baseada nos conceitos descritos nos capítulos anteriores. Este capítulo descreve os resultados obtidos com os estudos feitos para o desenvolvimento deste protótipo, e também, as dificuldades encontradas ao longo do desenvolvimento para a conclusão deste projeto.

4.1 – Projeto desenvolvido

A idéia era de construir um protótipo utilizando Software Livre. Para isso foram utilizadas as seguintes ferramentas:

- ✓ Sistema operacional: Linux – Conectiva 10, [14];
- ✓ Linguagem de desenvolvimento WEB: PHP, [34,35];
- ✓ Servidor HTTP: Apache, [38];
- ✓ Banco de Dados: MySQL [36];
- ✓ Pacote SNMP: Net-SNMP [14].

Para a elaboração do projeto, todas as tarefas realizadas durante o projeto estão descritas a seguir.

4.1.1 – Sistema operacional

Para iniciar a preparação do projeto foi instalado o Linux – Conectiva 10 como sistema operacional. O tipo de instalação realizada foi a padrão. Foram necessárias algumas configurações, como por exemplo, a configuração da rede. De uma forma geral, a instalação foi simples, não ocorrendo problemas.

A figura 4.1 mostra a tela principal do Conectiva.

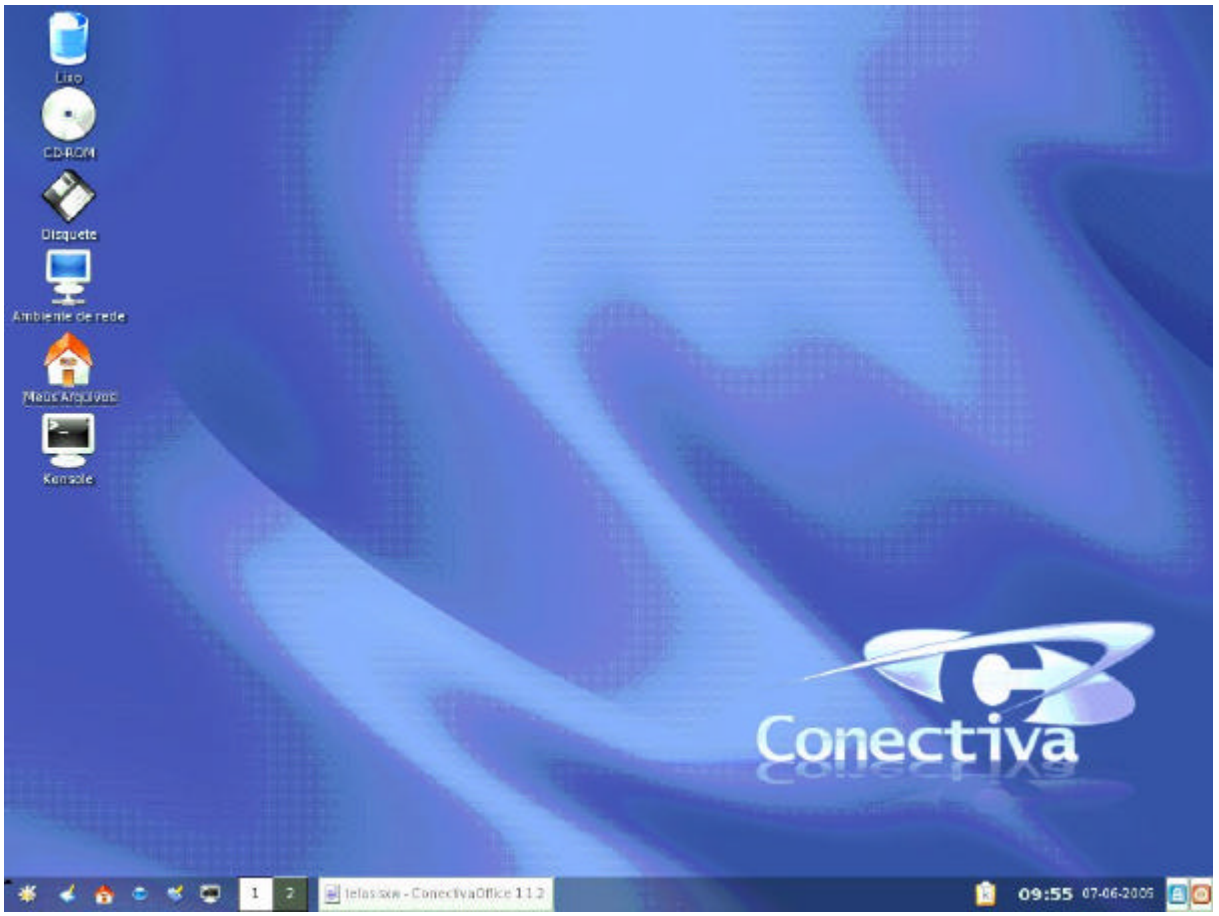

Figura 4.1 – Área de trabalho do Conectiva 10

4.1.2 – Pacote SNMP

Considerado o principal ponto deste projeto, foi instalado o pacote SNMP da Net-SNMP, disponível também no próprio Linux – Conectiva 10. Este pacote implementa os

comandos disponíveis no SNMP. Os comandos servem para requisitar ou gravar informações nos equipamentos gerenciados através dos agentes. Como exemplo, o comando *snmpwalk* é um dos comandos deste pacote. São comandos simples, assim como o próprio SNMP, mas que utilizados em conjunto e baseados em profundos estudos das informações disponíveis nas MIBs, podem gerar poderosos sistemas de gerenciamento.

Após a instalação deste pacote, o computador estava apto a ser um gerente e também um agente. Para a função de gerente, os comandos estavam prontos para serem utilizados. Para a função de agente, foi editado o arquivo de configuração *snmpd.conf*.

O comando principal utilizado para obtenção dos objetos gerenciados do agente de um dispositivo de rede foi o *snmpwalk*.

4.1.3 – Análise das informações disponíveis nas MIBs

Com os comandos SNMP disponíveis, foi possível obter os objetos gerenciados dos dispositivos para análise. Foram utilizados alguns dispositivos como um roteador Cisco 2500, servidores de rede e computadores.

Com o comando *snmpwalk*, foi obtido o acesso aos objetos de cada um destes dispositivos. As repostas deste comando trouxeram todas as informações disponíveis nos equipamentos.

A partir dessas informações realizou-se um estudo dos objetos e foram selecionados os que estariam relacionados com gerenciamento de falhas de hardware, conforme a seção 3.2.

4.1.4 – Desenvolvimento do código fonte

Após as instalações do sistema operacional e do pacote SNMP, e em seguida, as análises das informações da MIB através da comunicação entre o gerente e o agente, era necessário a criação de um ambiente gráfico.

O próximo passo para a materialização do protótipo seria a construção do código fonte. Foi utilizado o PHP como linguagem de desenvolvimento.

De posse do comando *snmpwalk* e dos objetos gerenciados, conforme a seção 3.2, já era possível a criação de um programa. No programa, utilizou-se uma seqüência de comandos *snmpwalk* nos objetos selecionados.

4.1.5 – Instalação de um servidor WEB

Este software é a que dá o suporte local na Estação de Gerenciamento para os acessos das páginas HTML e PHP. São através destas páginas que os usuários da aplicação obterão acessos às informações do gerenciamento em questão.

Para esta tarefa, foi utilizado servidor Apache, reconhecido como um excelente servidor WEB. A figura 4.2 mostra que o Apache foi instalado corretamente.

Figura 4.2 – Servidor Web Apache instalado e funcionando corretamente.

4.1.6 – Criação do banco de dados

Os dados lidos e coletados, conforme a seção 3.2, nos objetos do roteador foram gravados em um Banco de Dados que faz parte da arquitetura do protótipo.

Este Banco de Dados é necessário pelos seguintes fatores:

- ✓ Para os objetos cujos dados sejam do tipo "Counter32", por exemplo, é necessário saber o valor anterior da leitura para que seja possível fazer a diferença entre os dois valores e saber o valor do intervalo;
- ✓ Manter o histórico de leituras para geração de gráficos dos tráfegos das interfaces;
- ✓ Maior segurança e confiabilidade na manutenção das informações.

4.1.7 – Criação das páginas e geração dos gráficos

Esta é uma das tarefas mais importantes do projeto. As páginas são a interface com as informações do gerenciamento.

Este trabalho, procurou disponibilizar a forma de visualização das informações e os gráficos das interfaces do equipamento gerenciado. Foram implementadas páginas HTML onde são mostradas as informações do dispositivo.

4.1.8 – Arquitetura funcional do protótipo

Na figura 4.3, é mostrada a arquitetura do gerenciamento e o fluxo lógico do funcionamento do sistema.

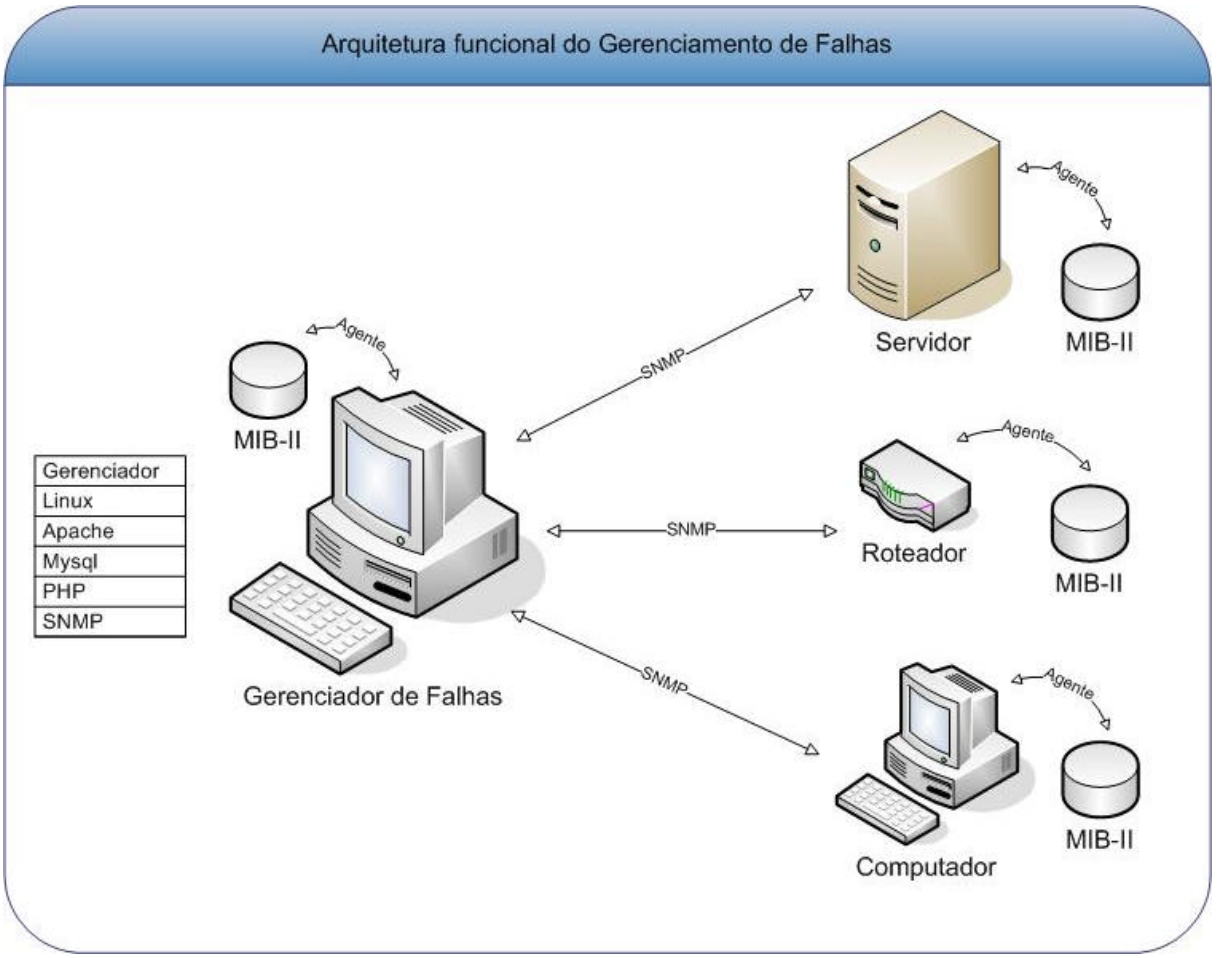

Figura 4.3 – Arquitetura funcional do Gerenciamento de Falhas

O Gerenciador de Falhas, ou Gerente, executando o Linux, o Apache, o Mysql, o PHP e o SNMP, faz consultas e requisições, através do comando *snmpwalk*, aos dispositivos de rede utilizando o protocolo de gerenciamento SNMP. Estes dispositivos podem ser: um servidor, um roteador ou um computador. Cada dispositivo possui um Agente que busca na MIB-II as informações solicitadas pelo Gerenciador. Na seção 4.2, será mostrado o funcionamento do gerenciador.

4.2 – Funcionamento do Gerenciador

Após todos os estudos, implementações e processos executados para a criação do protótipo, verificou-se o funcionamento do gerenciador, o qual será mostrado nesta seção.

4.2.1 – Interface do Gerenciador

Foi criada a interface do gerenciador para a utilização simples e prática do gerenciamento. Essa interface contém o título do projeto, o campo para a entrada do número IP do dispositivo que se quer gerenciar, o botão consulta que inicia a busca e o botão limpar para limpar a tela e os campos.

A figura 4.4 mostra a interface do Gerenciador.

Figura 4.4 – Interface principal do Gerenciador

4.2.2 – Serviços e Ferramentas

Para o correto funcionamento do Gerenciador, é necessário que todos os serviços e aplicações estejam habilitados e inicializados.

Na figura 4.5, observa-se que os serviços HTTP, MySQL e o SNMP estão funcionando.

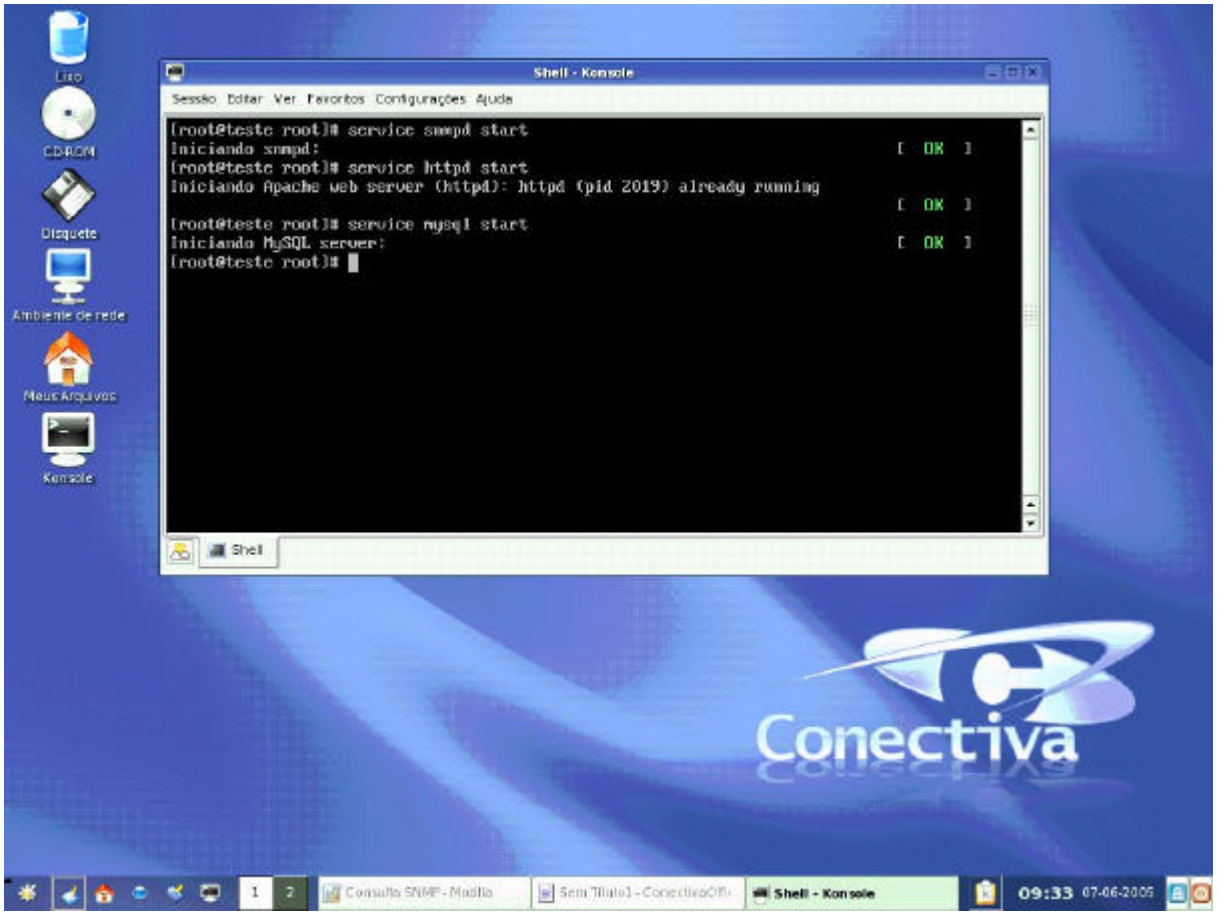

Figura 4.5 – Serviços essenciais inicializados e em funcionamento.

4.2.3 – Comando SNMPWALK

A principal operação SNMP utilizada no Gerenciador é o *snmpwalk*. Através deste comando é feita a busca dos objetos gerenciados.

A sintaxe deste comando é:

snmpwalk –O*[opções]* –c *[comunidade]* –v *[versão do protocolo]* ip_host grupo_objeto

Exemplos:

snmpwalk –Os –c public –v 2c 10.142.0.9 system

snmpwalk –Os –c public –v 1 200.202.142.254 interfaces.ifTable.ifEntry.ifSpeed

As figuras 4.6 e 4.7 mostram a comunicação SNMP estabelecida com os agentes.

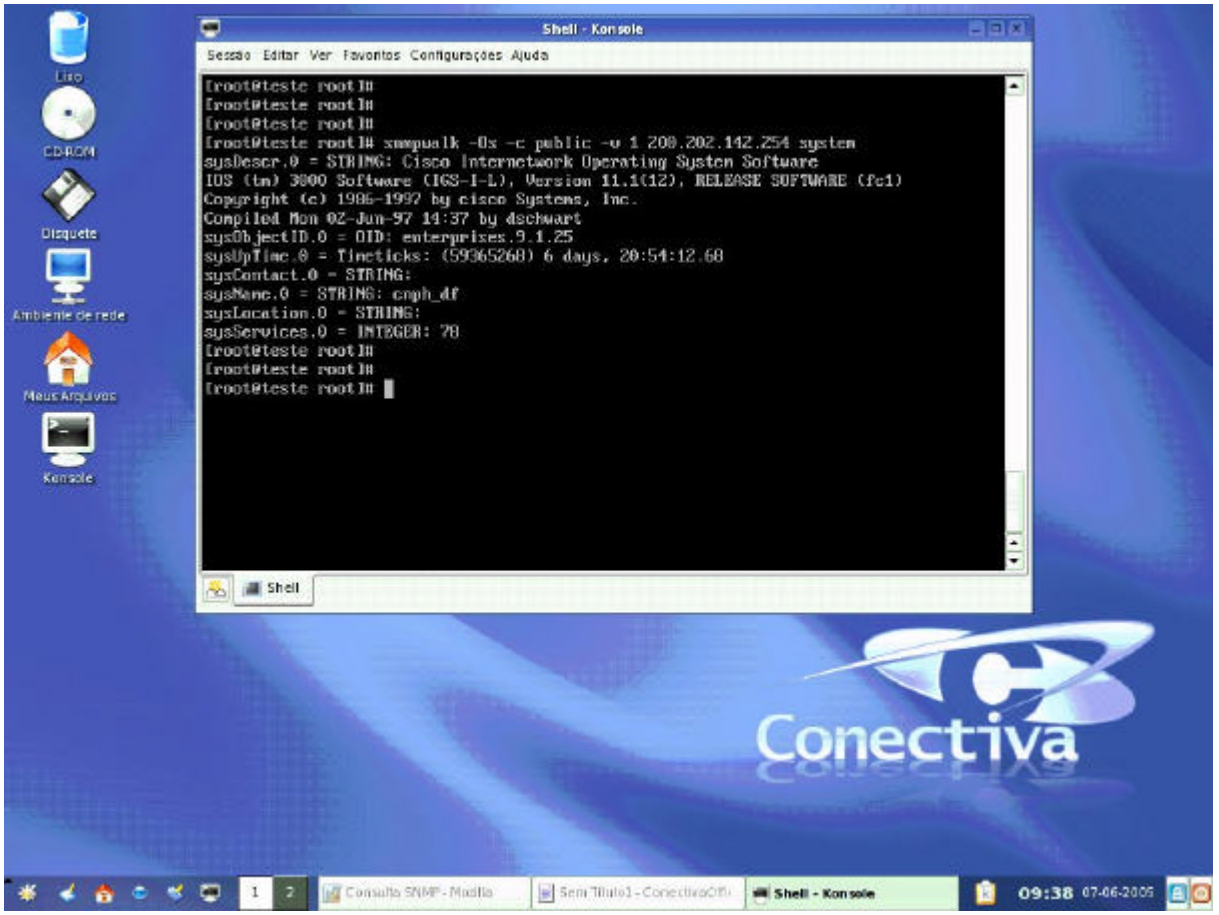

Figura 4.6 – Comunicação com um roteador Cisco com SNMP v1

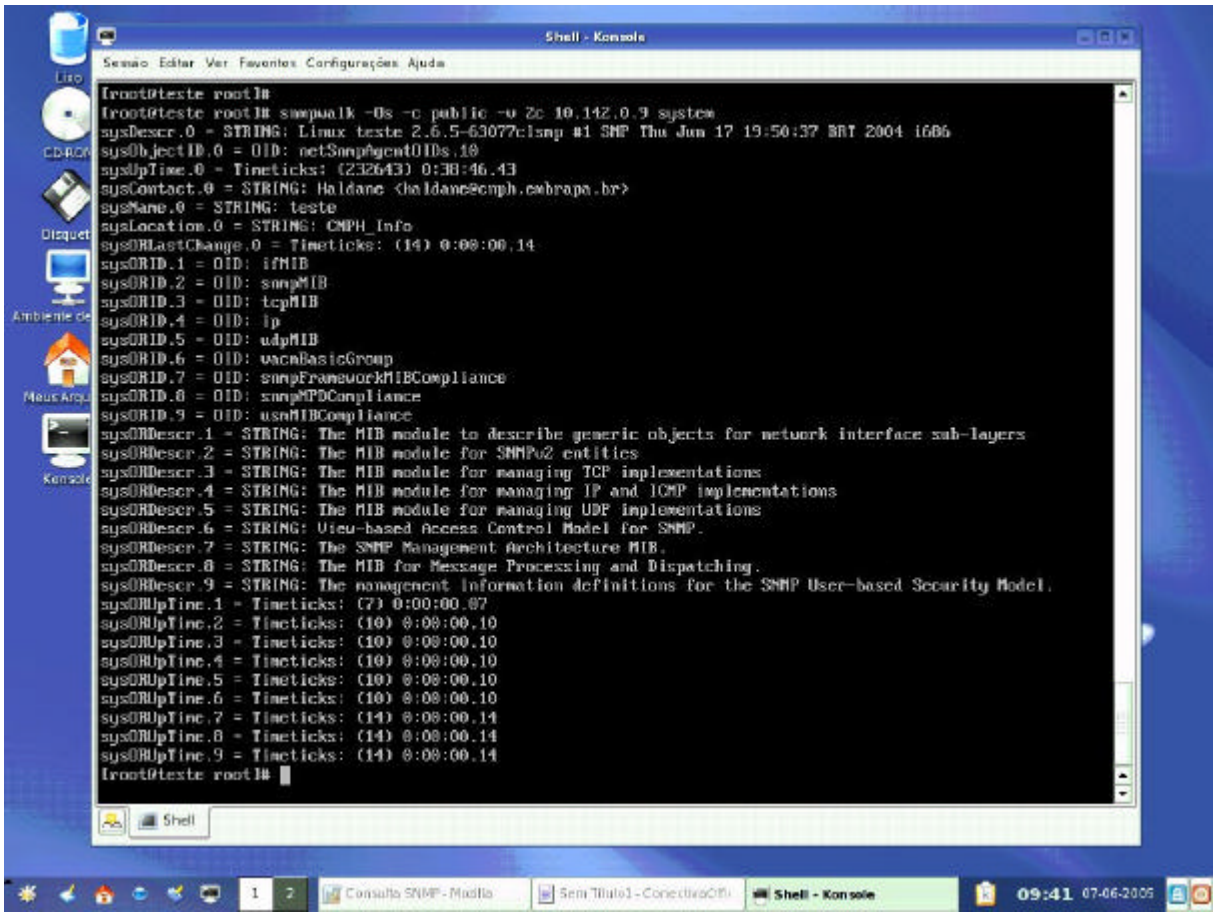

Figura 4.7 – Comunicação com o agente local com SNMP v2

4.2.4 – Gerenciando um dispositivo

Para validar o Gerenciador foram feitos testes e simulações. O funcionamento foi satisfatório. Através das figuras abaixo, podemos perceber que são obtidas várias informações necessárias para este tipo de gerenciamento. A seguir, os passos executados e os resultados gerados serão expostos.

Quando a comunicação entre o gerente e o agente é estabelecida, a resposta mostra-se ativa e, em seguida, são apresentados os objetos relacionados ao agente.

A figura 4.8 mostra o gerenciamento de um roteador Cisco.

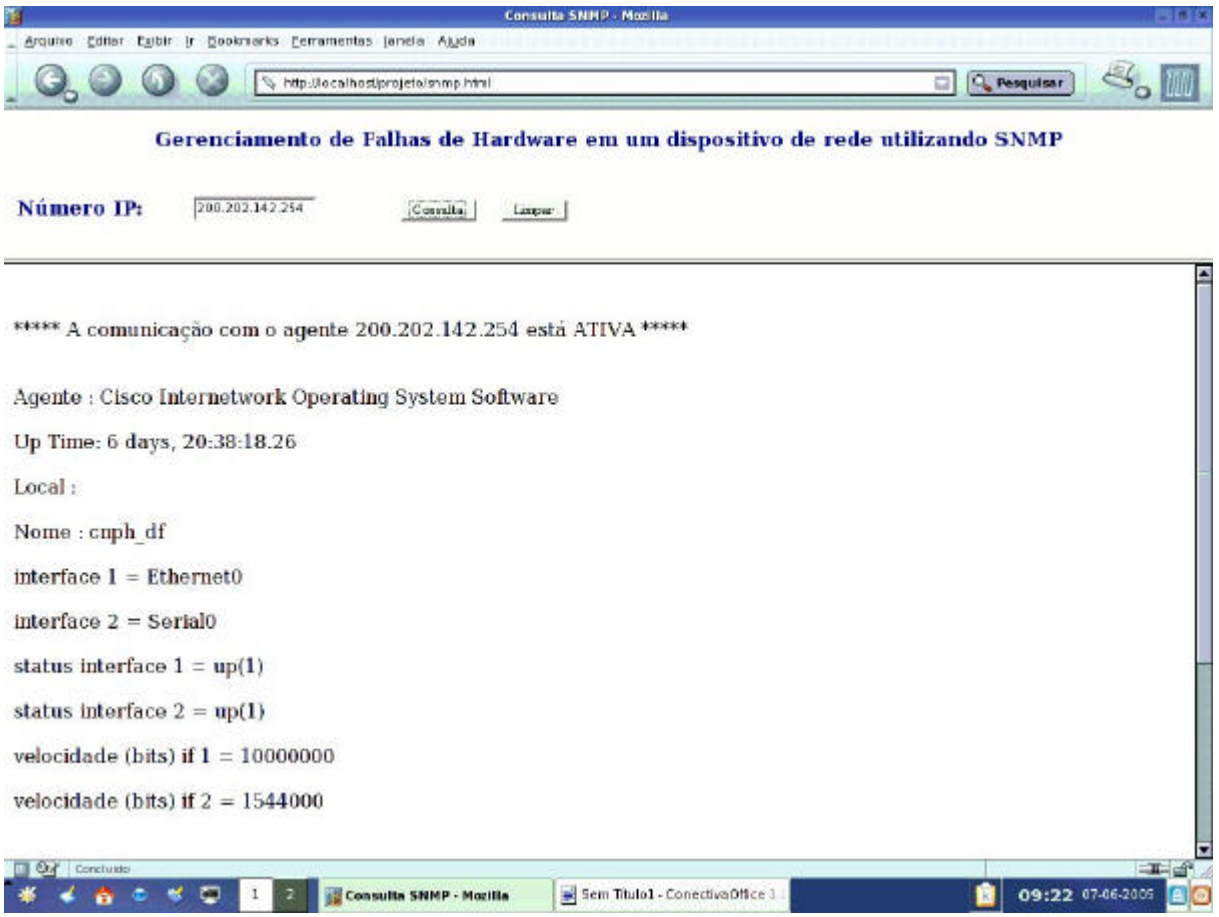

Figura 4.8 – Visualização da gerência de um roteador.

Caso não seja estabelecida a comunicação com o agente, a resposta será uma falha de comunicação. A figura 4.9 mostra uma falha ocorrida na tentativa de gerenciar um dispositivo.

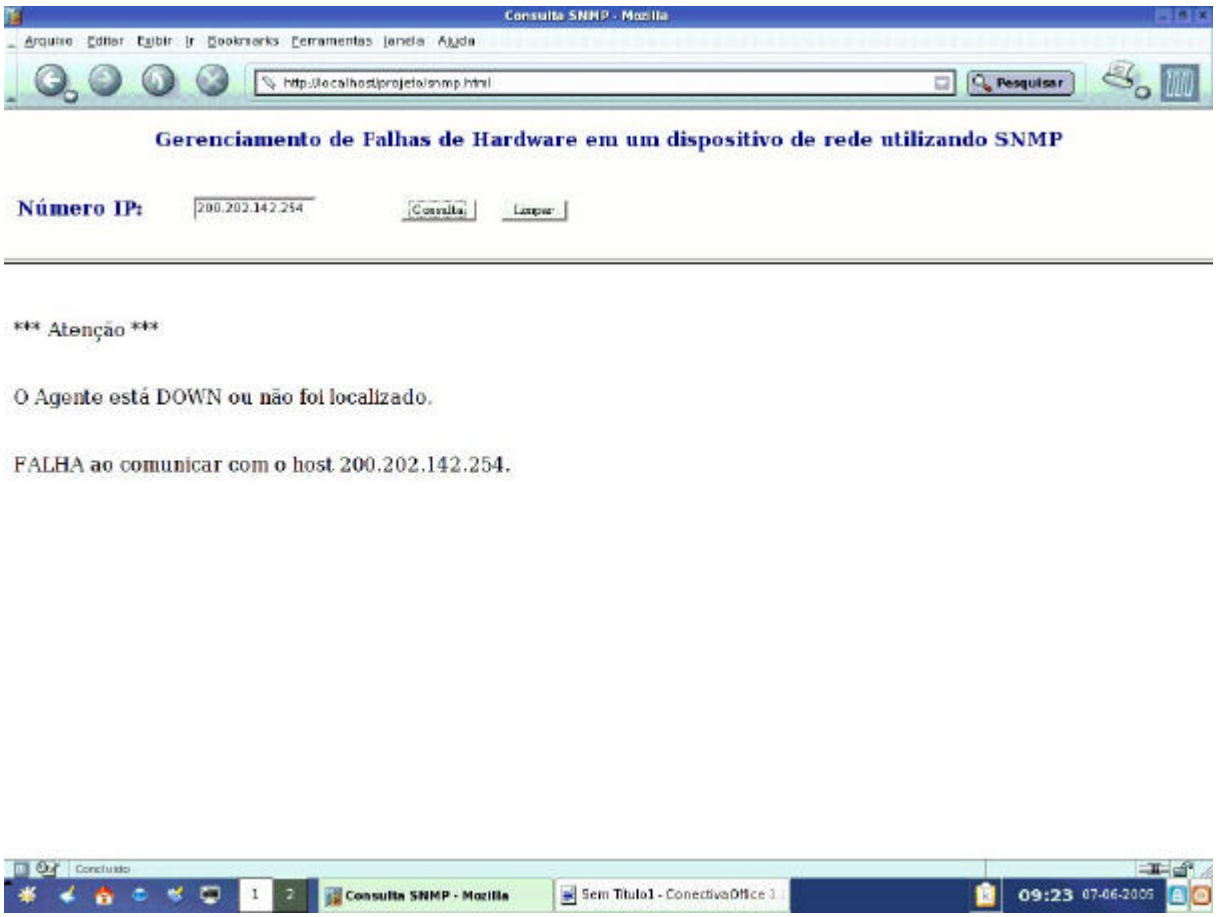

Figura 4.9 – Falha na comunicação com o agente

Em alguns casos, tem-se a opção de visualização de gráficos. No caso de estar gerenciando um roteador, é fundamental observar todo o tráfego das interfaces deste equipamento.

Com esta opção, pode-se identificar uma falha como, por exemplo, verificando os momentos em que houve uma queda, chegando a zero, na quantidade de *bytes* trafegados.

As figuras 4.10, 4.11, 4.12 e 4.13, mostram as duas interfaces do roteador e o tráfego, tanto de entrada como de saída, de cada uma delas. Observa-se, que em dois determinados momentos, ocorreram falhas no equipamento. Mais especificamente, quando os tráfegos atingiram a quantidade zero de *bytes* trafegados.

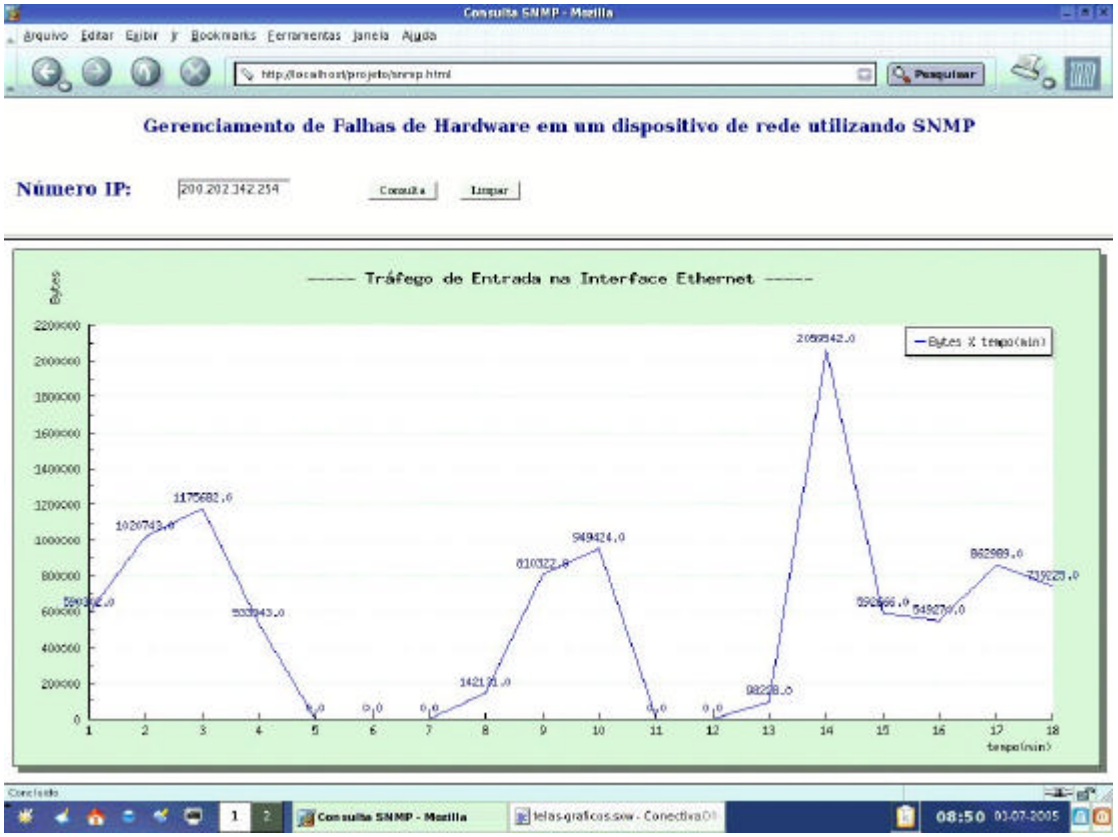

Figura 4.10 – Tráfego de entrada na interface Ethernet

Figura 4.11 – Tráfego de saída na interface Ethernet

Figura 4.12 – Tráfego de entrada na interface Serial

Figura 4.13 – Tráfego de saída na interface Serial

Para a geração dos gráficos foi necessário a utilização do banco de dados MySQL. As informações de gerenciamento eram armazenadas para posterior manipulação e tratamento dos dados com a finalidade da análise através de gráficos.

A figura 4.14 mostra uma base de dados colhida em um determinado intervalo de tempo em um dispositivo (roteador).

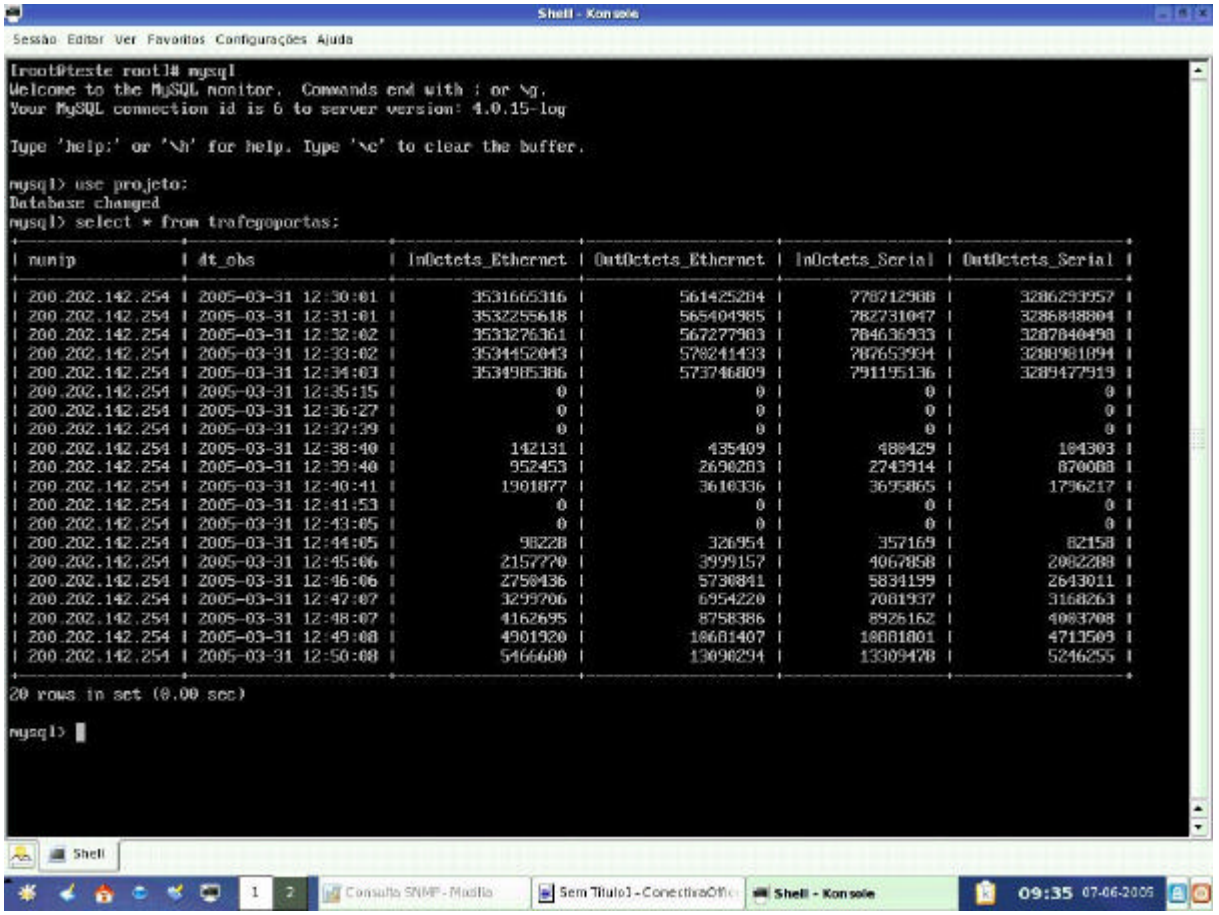

Figura 4.14 – Base de dados no MySQL

Através desta seção apresentada, a principal finalidade era mostrar de forma simples e objetiva o funcionamento do Gerenciador de Falhas de Hardware em um dispositivo de rede.

4.3 – Dificuldades encontradas

Como em todo projeto, não poderiam faltar as dificuldades. Desde o momento da escolha do tema até a conclusão do protótipo aconteceram momentos difíceis. Mas, como em todo trabalho, temos que vencer os obstáculos para conseguirmos realizar os planos propostos. Neste momento, descobriu-se o quanto era importante a conclusão do mesmo.

A seguir, as dificuldades encontradas:

- ✓ Material específico sobre gerenciamento de falhas;
- ✓ Utilização e aprendizado do Linux;
- ✓ A comunicação entre o gerente e o agente utilizando SNMP;
- ✓ Aprender e inserir no protótipo as ferramentas PHP, MySQL e Apache.

Apesar da Internet ser uma ótima fonte para buscas de trabalhos e referências, no caso de gerenciamento de falhas não foi obtido tanto sucesso. Livros também foram pesquisados. Encontram-se diversos conteúdos sobre gerenciamento, mas especificamente sobre falhas, há poucas bibliografias.

Para a realização desta parte específica foram selecionados diversos artigos e compilados para que pudesse ser estudado o assunto. De posse deste conteúdo foram feitos estudos e elaborados os entendimentos sobre gerenciamento de falhas de hardware. Uma das coisas importantes foram as consultas realizadas às pessoas da área de redes. Esclarecer dúvidas com técnicos da área continua sendo uma grande fonte de conhecimentos e aprendizado. Isso foi de grande valia para o desenvolvimento do projeto.

A falta de experiência com o sistema operacional Linux motivou os estudos posteriores e tornou-se um passo importante para vencer os outros obstáculos que viriam. A princípio, é um tanto quanto complicado utilizar o Linux. Principalmente por ser muito diferente do Windows da Microsoft. Como tudo, para aprendermos algo é necessário utilizarmos, mesmo com dificuldades. A partir do uso mais contínuo, foram adquiridos conhecimentos e eliminadas as barreiras.

A comunicação entre o gerente e o agente utilizando SNMP foi a maior dificuldade encontrada. Isso porque, era o ponto chave para o andamento do projeto. Com

muito estudo, pesquisa, instalações e testes, chegou-se ao funcionamento. Foi estabelecida a forma que era necessária para a comunicação SNMP entre o gerente e o agente.

Faltava mostrar e visualizar esta comunicação. Para isso, era necessário a construção de um ambiente propício para isso. Foi aí que se necessitou de uma linguagem de programação Web, um servidor HTTP e um banco de dados. Para não ser diferente, mais um obstáculo. Primeiro instalar o Apache, em seguida utilizar o PHP para construir os algoritmos e por fim disponibilizar o banco de dados através do MySQL. Ambas as ferramentas tiveram que ser estudadas.

5 – Conclusão

Os estudos teóricos e práticos adquiridos sobre gerenciamento de redes revelam que gerenciar uma rede implica realizar tarefas com o objetivo de monitorar e controlar seus recursos computacionais. Essa definição é bastante utilizada e aceita no meio acadêmico, muito embora seja uma forma muito simplista de definir toda a complexidade que há por trás da tarefa de gerenciar uma rede.

Ao elaborar e desenvolver este projeto foi possível observar que a tarefa de gerenciar uma rede e mantê-la funcionando em condições estáveis e níveis aceitáveis é uma tarefa complexa e difícil, que exige um elevado conhecimento da rede que está sendo gerenciada além do conhecimento das ferramentas utilizadas. Por isso, é necessário um gerenciamento completo e profundo da rede, incluindo todas as áreas funcionais de gerenciamento.

O gerenciamento de falhas é uma parte muito importante da gerência de redes, porque mantém a rede operacional e disponível no sentido de detectar, localizar e corrigir as falhas que estão acontecendo e, também, até mesmo as que vão surgir, dependendo da situação.

O trabalho proposto e desenvolvido teve como objetivo, criar um protótipo para gerenciar falhas de hardware em um dispositivo da rede.

Ao analisar o desenvolvimento do gerenciador, pode-se dizer que as ferramentas utilizadas foram adequadas ao objetivo do projeto. A comunidade Software Livre é ideal para trabalhos que necessitem de liberdade de criação, transparência de processos, gratuidade e do apoio de outros participantes da comunidade tecnológica.

Foi muito proveitoso adquirir conhecimentos das ferramentas utilizadas e aprender a usar parte de suas flexibilidades para conseguir o desejado, como, por exemplo, a simulação de um gerenciamento de roteador.

Sobre o produto gerado, pode-se dizer que o resultado foi uma ferramenta de simples manuseio, objetiva nas informações e muito prática para este tipo de gerenciamento.

Como sugestão para trabalhos futuros, seria interessante a construção de uma ferramenta mais aprofundada que focasse o gerenciamento de redes em todas as áreas funcionais de gerenciamento, conforme a seção 2.4. Além disso, poderia ser criada uma interface com mais recursos, como por exemplo, gerenciar automaticamente vários dispositivos ao mesmo tempo, dentro de intervalos pré-estabelecidos.

Como resultado final, o Gerenciador de Falhas de Hardware trouxe um rico aprendizado na área de redes de computadores que será, certamente, utilizado na minha carreira profissional.

6 – Referências bibliográficas

[Ammirabile, 2004] Ammirabile, Enriane K. e Giudice, Ricardo D., Detecção, Diagnóstico de Problemas e Procedimentos para uma Gerência Pró-ativa em uma rede WAN, Brasília, 2004.

[Azambuja, 2001] Azambuja, Marcelo C., PSWeM, Porto Alegre, 2001.

[Azambuja, 2000] Azambuja, Marcelo C., Silveira, Jorge G., Meirelles, Luiz F., WebMan, PUCRS, 2000.

[Comer, 1998] Comer, Douglas E., Interligação em rede com TCP/IP, vol.1, Campus, 1998.

[Dias, 2002] Dias, Beethovem Z. e Júnior, Nilton A., Protocolo de Gerenciamento SNMP, CBPF, 2002.

[Kurose, 2003] Kurose, James F. e Ross, Keith W., Redes de Computadores e a Internet, Addison Wesley, 2003.

[Metropoa, 2002] Metropoa, Laboratório, FreeNMS, PUC-RS, Porto Alegre, 2002.

[Sacks, 2003] Sacks, Anelise G., Sistema de gerenciamento de redes e processos através de computadores portáteis via Bluetooh, UFRJ, julho, 2003

[Salgues, 2004] Salgues, Joelson L. e Freitas, Wellita L., Gerência de Redes Corporativas, Brasília, 2004.

[Soares, 1995] Soares, L. F. e outros. Redes de Computadores: das LANs, MANs e WANs às redes ATM. Rio de Janeiro: Campus, 1995.

[Stallings, 1999] Stallings, William, SNMP, SNMPv2, SNMPv3 and RMON 1 and 2, Third Edition, Addison Wesley, 1999.

[Tanenbaum, 1995] Tanenbaum, Andrew S. Redes de Computadores. Rio de Janeiro, RJ: Campus, 1995.

[Weber, 1997] Weber, Taisy S., Tolerância a Falhas, UFRGS, 1997.

Sites na Internet:

[1] Pedro Sergio Nicolletti, http://www.dsc.ufcg.edu.br/~peter/, acessado em 13 de junho de 2005.

[2] Lisandro Zambenedetti Granville, http://www.inf.ufrgs.br/granville/, acessado em 13 de junho de 2005.

[3] Roteadores SNMP, http://www.pop-pr.rnp.br/tiki-index.php?page= Roteadores+SNMP, acessado em 13 de junho de 2005.

[4] Instalando e configurando SNMP e MRTG no Linux, Márcio Araújo Lopes, http://www.slacklife.com.br/article.php?sid=1451, acessado em 13 de junho de 2005.

[5] Linux Conectiva 10, http://www.conectiva.com.br/cpub/pt/principal/index.php, acessado em 13 de junho de 2005.

[6] Criando gráficos com a classe JPGraph, Fábio Berbert de Paula, http://phpbrasil.com/articles/article.php/id/411, acessado em 13 de junho de 2005.

[7] Gerenciamento de redes utilizando o protocolo SNMP, Italo Marcelo de O. Costa, http://www.htmlstaff.org/redes/redes22.php, acessado em 13 de junho de 2005.

[8] Introdução ao Modelo de Referência SNMP, Meirelles, L.F.T., setembro, 1997, http://redes.ucpel.tche.br/documentos/snmp/, acessado em 13 de junho de 2005.

[9] Redes de Computadores e suas Aplicações, http://penta2.ufrgs.br/, acessado em 13 de junho de 2005.

[10] Simpósio Brasileiro de Redes de Computadores – SBRC2004, http://www.sbrc2004.ufrgs.br/, acessado em 13 de junho de 2005.

[11] Introdução a Gerenciamento de Redes TCP/IP, NewsGeneration, RNP, http://www.rnp.br/newsgen/9708/n3-2.html, acessado em 13 de junho de 2005.

[12] Gerenciamento de Redes TCP/IP – continuação, NewsGeneration, RNP, http://www.rnp.br/newsgen/9712/gerencia.html, acessado em 13 de junho de 2005.

[13] Net-SNMP, www.net-snmp.org , acessado em 13 de junho de 2005.

[14] http://net-snmp.sourceforge.net/, acessado em 13 de junho de 2005.

[15] Apache Software Foundation, http://www.apache.org/, acessado em 13 de junho de 2005.

[16] MySQL, http://www.mysql.com/, acessado em 13 de junho de 2005.

[17] RFC's, http://www.rfc-editor.org/rfcsearch.html, acessado em 13 de junho de 2005.

[18] RFC's, http://www.ietf.org/rfc.html, acessado em 13 de junho de 2005.

[19] RFC's, http://www.faqs.org/rfcs/, acessado em 13 de junho de 2005

[20] FreeNMS, http://www.freenms.org/, acessado em 13 de junho de 2005.

[21] http://www.snmplink.org/, acessado em 13 de junho de 2005.

[22] http://www.simpleweb.org/, acessado em 13 de junho de 2005.

[23] PHP, http://www.php.net/, acessado em 13 de junho de 2005.

[24] PHP, http://www.phpbrasil.com/, acessado em 13 de junho de 2005.

[25] Linux Online, http://www.linux.org/, acessado em 13 de junho de 2005.

7 – Anexos

Anexo A – Código fonte da tela principal do gerenciador

snmp.html

```
<html>
<head>
<title>Consulta SNMP</title>
</head>
  <frame name="menu" scrolling="no" noresize target="rtop" src="menu.html">
  <frameset rows="20%,*">
    <frame name="menu" target="rtop" src="menu.html" scrolling="no">
    <frame name="detalhe" src="branco.html" target="detalhe">
  </frameset>
  <noframes>
  <body>
  <p>This page uses frames, but your browser doesn't support them.</p>
  </body>
  </noframes>
</html>
```

branco.html

```
<html>
<head>
<title>Gerenciamento de Falhas de Hardware</title>
<meta name="GENERATOR" content="Microsoft FrontPage 3.0">
<base target="detalhe">
</head>
<body>
</body>
</html>
```

menu.html

```
<html>
<head>
<title>Gerenciamento de Falhas de Hardware</title>
<meta name="GENERATOR" content="Microsoft FrontPage 3.0">
<base target="detalhe">
</head>
<body bgcolor="#FFFFE0">
<form method="POST" action="consultaip.php">
  <div align="center"><center><p><font face="Arial" color="#000080"><strong>Gerenciamento de Falhas de Hardware em um dispositivo de rede utilizando SNMP</strong></font></p>
  </center></div>
  <br>
  <table border="0" width="585" cellspacing="1" cellpadding="2"
  bordercolor="#FFFFFF">
    <tr>
      <td width="144"><font color="#000080"><strong>Número IP:</strong></font></td>
      <td width="169"><input type="text" name="ip" size="15" maxlength="15"></td>
      <td width="255"><input type="submit" value="Consulta" name="B1" style="font-family:Arial">    <input
      type="reset" value="Limpar" name="limpa" style="font-family: Arial"
      onClick="javascript:window.open('branco.html','detalhe')"></td>
    </tr>
  </table>
</form>
</body>
</html>
```

consultaip.php

```
<html>
<head>
```

```
<title>Gerenciamento de falhas de Harware em um dispositivo de rede utilizando SNMP</title>
<base target="detalhe">
</head>
<body>
<br>
<?
$ip = $_REQUEST['ip'];
$saida = shell_exec("snmpwalk -Os -c public -v 1 $ip system.sysDescr.0");
$host=explode(" ",$saida);
if ($host[0] == "sysDescr.0")
   {
   echo "<br>***** A comunicação com o agente $ip está ATIVA *****<br><br><br>";
   $sysdescr = shell_exec("snmpwalk -Os -c public -v 1 $ip system.sysDescr.0");
   ereg("STRING: ([[:alpha:].0-9 #-:]{1,})",$sysdescr, $descr);
   echo "Agente : " . $descr[1] . "<br><br>";
   $sysuptime = shell_exec("snmpwalk -Os -c public -v 1 $ip system.sysUpTime");
   ereg("Timeticks: ([(0-9)]{1,}) ([0-9].{1,})",$sysuptime, $uptime);
   echo "Up Time: " . $uptime[2] . "<br><br>";
   $syslocation = shell_exec("snmpwalk -Os -c public -v 1 $ip system.sysLocation.0");
   ereg("STRING: ([[:alpha:]_]{1,})",$syslocation, $location);
   echo "Local  : " . $location[1]  . "<br><br>";
   $sysname = shell_exec("snmpwalk -Os -c public -v 1 $ip system.sysName.0");
   ereg("STRING: ([[:alpha:]_]{1,})",$sysname, $name);
   echo "Nome   : " . $name[1] . "<br><br>";
   $ifdescr1 = shell_exec("snmpwalk -Os -c public -v 1 $ip interfaces.ifTable.ifEntry.ifDescr.1");
   ereg("STRING: ([[:alpha:].[0-9]{1,})",$ifdescr1, $descr1);
   echo "interface 1 = " . $descr1[1] . "<br><br>";
   $ifdescr2 = shell_exec("snmpwalk -Os -c public -v 1 $ip interfaces.ifTable.ifEntry.ifDescr.2");
   ereg("STRING: ([[:alpha:].[0-9]{1,})",$ifdescr2, $descr2);
   echo "interface 2 = " . $descr2[1] . "<br><br>";
```

```
$ifoperstatus1 = shell_exec("snmpwalk -Os -c public -v 1 $ip interfaces.ifTable.ifEntry.ifOperStatus.1");
ereg("INTEGER: ([[:alpha:].(0-9)]{1,})",$ifoperstatus1, $operstatus1);
echo "status interface 1 = " . $operstatus1[1] . "<br><br>";
$ifoperstatus2 = shell_exec("snmpwalk -Os -c public -v 1 $ip interfaces.ifTable.ifEntry.ifOperStatus.2");
ereg("INTEGER: ([[:alpha:].(0-9)]{1,})",$ifoperstatus2, $operstatus2);
echo "status interface 2 = " . $operstatus2[1] . "<br><br>";
$ifspeed1 = shell_exec("snmpwalk -Os -c public -v 1 $ip interfaces.ifTable.ifEntry.ifSpeed.1");
ereg("Gauge32: ([0-9]{1,})",$ifspeed1, $speed1);
echo "velocidade (bits) if 1 = " . $speed1[1] . "<br><br>";
$ifspeed2 = shell_exec("snmpwalk -Os -c public -v 1 $ip interfaces.ifTable.ifEntry.ifSpeed.2");
ereg("Gauge32: ([0-9]{1,})",$ifspeed2, $speed2);
echo "velocidade (bits) if 2 = " . $speed2[1] . "<br><br>";
?><form method="POST" action="grafconsultaip11.php"><br>
 <input type="submit" value="Tráfego de entrada if 1" name="traf" style="font-family:Arial">
</form><?
?><form method="POST" action="grafconsultaip12.php"><br>
 <input type="submit" value="Tráfego de saída if 1" name="traf" style="font-family:Arial">
</form><?
?><form method="POST" action="grafconsultaip21.php"><br>
 <input type="submit" value="Tráfego de entrada if 2" name="traf" style="font-family:Arial">
</form><?
?><form method="POST" action="grafconsultaip22.php"><br>
 <input type="submit" value="Tráfego de saída if 2" name="traf" style="font-family:Arial">
</form><?
}
else
```

```
    echo "<br>*** Atenção ***<br><br><br>O Agente está DOWN ou não foi localizado.
<br><br><br> FALHA ao comunicar com o host $ip. ";
?>
</body>
</html>
```

Anexo B – Código fonte da tela dos gráficos

grafconsultaip12.php

```
<?
require ("/usr/local/jpgraph/src/jpgraph.php");
require ("/usr/local/jpgraph/src/jpgraph_line.php");
require ("sqlbd.php");
$host = "localhost";
$base = "projeto";
$user = "root";
$pass = "";
$db = new SQLBD($host,$base,$user,$pass);
$db->connect();
$consulta = "select OutOctets_Ethernet from trafegoportas";
$db->query($consulta);
while ($db->movimenta())
{
  $saidas1[]=$db->valor["OutOctets_Ethernet"];
}
$db->close();
$ct = count($saidas1) - 2;
$guarda = 0;
for ($i=0; $i < $ct;$i++)
{
 if ($saidas1[$i] == 0 && $guarda == 0)
    {
```

```
    $guarda = $saidas[$i-1];
    }
  if ($saidas1[$i+1] <= $saidas1[$i])
    {
     $saidas[]=0;
  } else {
    if ($guarda != 0)
      {
        $saidas[]=$saidas1[$i+1] - $guarda;
        $guarda = 0;
    } else {
        $saidas[]=$saidas1[$i+1] - $saidas1[$i];
      }
      }
}
$origem = array ("Ethernet");
$grafico = new graph(1010,485,"auto");
$grafico->img->SetMargin(70,50,70,50);
$grafico->SetScale("textlin");
$grafico->SetShadow();
$lineplot = new LinePlot($saidas);
$lineplot->value->show();
$grafico->Add($lineplot);
$grafico->title->Set('
----- Tráfego de Saída na Interface Ethernet -----');
$grafico->xaxis->title->Set("tempo(min)");
$grafico->yaxis->title->Set("                                                    Bytes");
$lineplot->SetColor("green");
$lineplot->SetLegend("Bytes X tempo(min)");
$lineplot->SetWeight(1);
$grafico->Stroke();
?>
```