

UNIVERSIDADE SAGRADO CORAÇÃO

GUSTAVO MARTINELI SANCHES

**ANÁLISE DOS RECURSOS DOS SOFTWARES LIVRES
CACTI E NAGIOS QUE UTILIZAM O PROTOCOLO SNMP
PARA GERENCIAMENTO DE REDES**

BAURU
2011

UNIVERSIDADE SAGRADO CORAÇÃO

GUSTAVO MARTINELI SANCHES

**ANÁLISE DOS RECURSOS DOS SOFTWARES LIVRES
CACTI E NAGIOS QUE UTILIZAM O PROTOCOLO SNMP
PARA GERENCIAMENTO DE REDES**

Trabalho de conclusão de curso apresentado à Universidade Sagrado Coração, para a obtenção do título de bacharel em Ciência da Computação, sob orientação do prof. Esp. Henrique Pachioni Martins.

BAURU
2011

S2111a	<p data-bbox="548 1247 886 1274">Sanches, Gustavo Martineli</p> <p data-bbox="548 1310 1278 1436">Análise dos recursos dos softwares livres Cacti e Nagios que utilizam o protocolo SNMP para gerenciamento de redes / Gustavo Martineli Sanches -- 2011.</p> <p data-bbox="599 1444 683 1472">77f.: il.</p> <p data-bbox="599 1507 1224 1535">Orientador: Prof. Esp. Henrique Pachioni Martins</p> <p data-bbox="548 1570 1278 1667">Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade Sagrado Coração – Bauru – SP.</p> <p data-bbox="548 1703 1278 1759">1. Gerência de redes. 2. SNMP. 3. Softwares livres. 4. Nagios. 5. Cacti. I. Martins, Henrique Pachioni. II.</p>
--------	--

GUSTAVO MARTINELI SANCHES

**ANÁLISE DOS RECURSOS DOS SOFTWARES LIVRES CACTI E
NAGIOS QUE UTILIZAM O PROTOCOLO SNMP PARA
GERENCIAMENTO DE REDES**

Trabalho de Conclusão de Curso apresentado ao Centro de Ciências Exatas e Sociais Aplicadas da Universidade Sagrado Coração como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação, sob orientação do Prof. Esp. Henrique Pachioni Martins.

Banca examinadora:

Prof. Esp. Henrique Pachioni Martins
Universidade Sagrado Coração

Prof. Esp. Andre Luiz Ferraz Castro
Universidade Sagrado Coração

Prof. Dr. Elvio Gilberto da Silva
Universidade Sagrado Coração

Bauru, 13 de dezembro de 2011.

Dedico este trabalho aos meus pais, avós e irmão por todo o apoio depositado em mim ao longo dos estudos durante minha carreira acadêmica. Em especial, dedico à minha namorada Bruna, que esteve ao meu lado proporcionando força e carinho em todos os momentos necessários.

AGRADECIMENTOS

Agradeço primeiramente a Deus, que sempre abençoou e iluminou meus passos mostrando os caminhos certos a seguir, além de me proteger de todo mal e conceder uma família especial em minha vida.

Agradeço aos meus pais e meu irmão por todo apoio e incentivo nessa caminhada universitária em busca dos meus sonhos.

Agradeço também aos meus avós que me educaram e forneceram a base de toda minha educação desde os primórdios dos meus estudos.

Sem esquecer-se dos meus amigos, professores da Universidade Sagrado Coração e todos aqueles que de uma forma ou de outra colaboraram e compartilharam momentos únicos e inesquecíveis à minha formação profissional.

LISTA DE ILUSTRAÇÕES

Figura 1 - Formas de Implantação do Módulo Tradutor.....	24
Figura 2 - Componentes do SNMP	29
Figura 3 - Estrutura da MIB.....	31
Figura 4 - Protocolo SNMP sobre a camada de transporte.....	34
Figura 5 - Formato das mensagens SNMP.....	35
Figura 6 - Monitoramento de alguns serviços pelo Cacti.....	40
Figura 7 - Monitoramento de alguns serviços pelo Nagios.....	42
Figura 8 - Tela de Login do Cacti	47
Figura 9 - Tela inicial do Cacti.	48
Figura 10 - Gráfico do uso de memória gerado pelo Cacti.....	49
Figura 11 - Gráfico da média de carregamento gerado pelo Cacti.....	50
Figura 12 - Gráfico de usuários logados gerado pelo Cacti.	51
Figura 13 - Gráfico de processos gerado pelo Cacti.....	52
Figura 14 - Tela para gerenciar o computador que está sendo monitorado.....	53
Figura 15 - Dispositivos sendo monitorados pelo Cacti.	54
Figura 16 - Consulta de dados pelo Cacti.	55
Figura 17 - Visualização de modelos gráficos, host e de dados gerado pelo Cacti...	56
Figura 18 - Configurações gerais disponíveis pelo Cacti.....	57
Figura 19 - Usuários monitorados pelo Cacti.....	58
Figura 20 - Tela de login do Nagios.	59
Figura 21 - Tela inicial do Nagios.....	60
Figura 22 - Resumo geral da performance de monitoramento do Nagios.....	61
Figura 23 - Recursos de monitoramento do Nagios.....	61
Figura 24 - Detalhamento de hosts e serviços do Nagios.	62
Figura 25 - Detalhamento de hosts e serviços do Nagios.	63
Figura 26 - Detalhamento de hosts e serviços do Nagios.	63
Figura 27 - Relatórios do Nagios – Passo 1.....	64
Figura 28 - Relatórios do Nagios – Passo 2.....	65
Figura 29 - Relatórios do Nagios – Passo 3.....	66
Figura 30 - Relatórios do Nagios – Detalhamento.....	67
Figura 31 - Gerenciamento de alertas.	68
Figura 32 - Registro de eventos do Nagios.....	69

Figura 33 - Informações dos processos realizados pelo Nagios.....	70
Figura 34 - Checagem de desempenho dos serviços.....	71
Figura 35 - Fila de agendamento de serviços.....	72

LISTA DE ABREVIATURAS E SIGLAS

ARP – Address Resolution Protocol
ASN.1 – Abstract Syntax Notation 1
AT&T – American Telephone and Telegraph
BSD – Berkeley Software Distribution
CGIs – Common Gateway Interfaces
CMIP – Common Management Information Protocol
CMOT – CMIP over TCP/IP
CPU – Central Processing Unit
DEC – Digital Equipment Corporation
EGP – External Gateway Protocol
GNMP – Government Network Management Profile
GNU – Gnu is Not Unix
GOSIP – Government OSI Profile
GPL – General Public License
GUI – Graphical User Interface
HEMS – High-Level Entity Management System
HMP – Host Management Protocol
IAB – Internet Architecture Board
IBM – International Business Machines
ICMP – Internet Control Message Protocol
IEC – International Electrotechnical Commission
IFIP – International Federation for Information Processing
IP – Internet Protocol
ISO – International Organization for Standardization
LAN – Local Area Network
MIB – Management Information Base
MySQL – My Structured Query Language
NIST – National Institute of Standards and Technology
NM Fórum – Network Management Fórum
OID – Object Identification
OSI – Open Systems Interconnection
PABX – Private Automatic Branch Exchange

PC – Personal Computer

PDU – Protocol Data Unit

PHP – Hypertext Preprocessor

RDLM – Remote Digital Line Module

RFC – Request For Comments

RMON – Remote Monitoring

RRD – Round Robin Database

SGMP – Simple Gateway Monitoring Protocol

SMI – Structure of Management Information

SNMP – Simple Network Management Protocol

SNMPv2 – Simple Network Management Protocol versão 2

SNMPv3 – Simple Network Management Protocol versão 3

TCP – Transmission Control Protocol

UDP – User Datagram Protocol

SUMÁRIO

1 INTRODUÇÃO	12
1.1 PROBLEMA.....	13
2 OBJETIVOS	14
2.1 OBJETIVO GERAL.....	14
2.2 OBJETIVOS ESPECÍFICOS	14
3 JUSTIFICATIVA	15
4 REVISÃO DE LITERATURA	16
4.1 GERENCIAMENTO DE REDES	16
4.1.1 Importância do gerenciamento de redes.....	18
4.1.2 Necessidade do gerenciamento de redes.....	20
4.1.3 Áreas da gerência	21
4.1.4 Gerência de configuração.....	21
4.1.5 Gerência de falhas	21
4.1.6 Gerência de desempenho	22
4.1.7 Gerência de segurança.....	22
4.1.8 Gerência de contabilidade	23
4.1.9 Histórico do gerenciamento de redes.....	23
4.2 PROTOCOLO SNMP (SIMPLE NETWORK MANAGEMENT PROTOCOL)	27
4.2.1 Principais objetivos do protocolo SNMP	28
4.2.2 Agente e gerente	28
4.2.3 MIB – <i>Management Information Base</i>	29
4.2.4 Estrutura da MIB.....	30
4.2.5 Pontos positivos e negativos do snmp	31
4.2.6 Operações do SNMP	32
4.2.7 Funcionamento do snmp	33
4.3 SOFTWARE LIVRE	35
4.4 CACTI	37
4.5 NAGIOS	40
5 METODOLOGIA	43
5.1 TIPO DE PESQUISA.....	43
5.2 MATERIAIS	43
5.3 PROCEDIMENTOS	44
6 RESULTADOS OBTIDOS	45
7 CONCLUSÃO	73
REFERÊNCIAS	76

RESUMO

A evolução tecnológica provoca um grande impacto na sociedade atualmente. A informação tem-se tornado objeto de significativa vantagem competitiva entre as empresas e organizações em seus investimentos e negócios; e é com base nisto que as redes de computadores estão em crescimento contínuo. Para manter um controle sobre a situação, surgiu o conceito de gerencia de redes, que visa maximizar a eficiência e produtividade sobre as informações que trafegam pela rede. Devido a essa grande necessidade de gerenciamento, foi inevitável que padrões de ferramentas fossem adotados e com isso surgiu o padrão SNMP. Porém, o gerenciamento se torna cada vez mais complexo devido à proporção com que as redes tornam-se maiores, avançadas e heterogêneas. Conseqüentemente é necessária a adoção de ferramentas automatizadas para a sua monitoração e controle. Mas essas mesmas ferramentas quase sempre são utilizadas incorretamente, isto é, possuem características inexploradas ou utilizadas de modo pouco eficiente. Para gerenciar um recurso, é necessário conhecê-lo muito bem e visualizar claramente o que este recurso representa no contexto da rede. Diante de tais fatos, este trabalho tem como objetivo realizar uma análise de dois *softwares* livres (Nagios e Cacti) baseados no protocolo SNMP para gerenciamento de redes, identificando os recursos oferecidos por ambos além de proporcionar uma visão mais detalhada e permitir que administradores de redes possam obter uma base de escolha para tais *softwares* de acordo com suas necessidades.

Palavras-chave: Gerencia de Redes, SNMP, *Softwares* Livres, Nagios, Cacti.

ABSTRACT

The technological progress causes a large impact on society today. Information has become an object of significant competitive advantage between companies and organizations in their investments and business, and on this basis is that computer networks are constantly growing. To maintain control over the situation, the concept of network management emerged, and this concept aims to maximize efficiency and productivity of the information through the network. Because of this tremendous need for management, it was inevitable that patterns were adopted. So, the SNMP standard came up. However, management becomes ever more complex due to the proportion which the networks have become larger, advanced and heterogeneous. Consequently, the adoption of automated tools for its monitoring and control were necessary. But these same tools are often used incorrectly, that is, have characteristics so unexplored or used inefficiently. In order to manage a resource, you must know him very well and clearly see what this feature represents in the context of the network. Faced with such facts, this research aims to conduct an analysis of two free softwares (Nagios and Cacti) based on SNMP protocol for network management, identifying the features offered by both in addition to providing a more detailed view and allow network administrators to gain a basis choice for such software according to their needs.

Key-words: Network Management, SNMP, Tools, Nagios, Cacti

1 INTRODUÇÃO

Na visão de Santos (2005), os avanços tecnológicos exercem hoje um grande impacto na sociedade. A informação tem-se tornado cada vez mais uma vantagem competitiva para as empresas e organizações em investimentos futuros. O fato é que, cada vez mais, as empresas, para se tornarem competitivas e sobreviverem no mercado, têm investido em tecnologia de informação, como a única forma de tornar seguro o processo decisório. E é nesse quadro que as redes de computadores se proliferam, encurtando as distâncias e diminuindo o tempo de resposta entre as transações de organizações por todo o mundo.

Santos (2005) ainda afirma que em decorrência das vantagens que as redes de computadores oferecem, o número e a extensão dessas estão em expansão contínua. À medida que as redes crescem em escala e extensão, dois fatores vão ficando mais evidentes: as redes, juntamente com seus recursos e aplicações, tornam-se cada vez mais indispensáveis para as organizações que as utilizam, e uma maior possibilidade de ocorrerem problemas, o que pode levar as redes a um estado de inoperância ou a níveis inaceitáveis de desempenho.

Para controlar tudo isso, surgiu o conceito de gerência de redes, que visa maximizar sua eficiência e produtividade.

Segundo Soares (1995, p. 434),

A monitoração do tráfego, do estado e do desempenho de uma estação da rede, assim como a monitoração do meio de transmissão e de outros sinais, é necessária para o gerenciamento da rede, de forma a possibilitar a detecção de erros, diagnoses e resoluções de problemas, tais como, falhas, diminuição do desempenho etc.

A fim de garantir certa qualidade dos serviços a seus usuários, é que as redes de computadores devem ser gerenciadas. Este gerenciamento envolve o monitoramento e o controle de recursos distribuídos com o objetivo de assegurar que sistemas de informação estejam operacionais e eficazes a todo instante.

Com esta crescente necessidade de gerenciamento, fez-se necessário que padrões para ferramentas fossem estabelecidos e, em resposta a esta necessidade surgiu o padrão SNMP.

Conforme lembra Santos (2005), o gerenciamento da rede realizado pelo protocolo *Simple Network Management Protocol* (SNMP), permite que uma ou mais

máquinas na rede sejam designadas gerentes da rede. Esta máquina recebe informações de todas as outras máquinas, chamadas agentes, e através do processamento destas informações pode gerenciar toda a rede e detectar facilmente problemas ocorridos.

No entanto, o gerenciamento tornou-se por si só complexo devido à proporção com que as redes tornaram-se maiores (extensão), avançadas (tecnologia) e heterogêneas (plataformas de *hardware* e *software* distintas).

Conseqüentemente o gerenciamento não pode ser realizado somente pelo esforço humano, ou seja, é necessária a adoção de ferramentas automatizadas para a sua monitoração e controle.

Por outro lado, como diz Santos (2005), essas mesmas ferramentas quase sempre são subutilizadas, isto é, possuem características inexploradas ou utilizadas de modo pouco eficiente. Para gerenciar um recurso, é necessário conhecê-lo muito bem e visualizar claramente o que este recurso representa no contexto da rede.

Diante de tais fatos, este trabalho tem como objetivo realizar uma análise de dois *softwares* livres (Nagios e Cacti) baseados no protocolo SNMP para gerenciamento de redes, identificando os recursos oferecidos por ambos além de proporcionar uma visão mais detalhada e permitir que gerentes de redes possam obter uma base de escolha para tais *softwares* de acordo com suas necessidades.

1.1 PROBLEMA

Quais recursos os *softwares* livres baseados no protocolo SNMP oferecem para o gerenciamento de redes?

2 OBJETIVOS

2.1 OBJETIVO GERAL

Analisar os recursos de *softwares* livres que utilizam o protocolo SNMP no gerenciamento de redes.

2.2 OBJETIVOS ESPECÍFICOS

- Identificar vantagens e desvantagens do uso de ferramentas livres para o gerenciamento de redes.

- Realizar análises dos recursos dos *softwares* livres a fim de se propor uma base de escolha para aqueles que procuram ferramentas com características voltadas ao protocolo SNMP.

3 JUSTIFICATIVA

As organizações públicas ou privadas detêm atualmente de redes de computadores de pequeno a grande porte em suas instalações. Muitas dessas organizações sequer conhecem os mecanismos para realizar um gerenciamento das informações em suas redes, onde dados importantes trafegam por seus mais variados dispositivos.

Atualmente, as redes de computadores e os seus recursos associados, tem se tornado fundamental e de tal importância para uma organização, que elas basicamente “não podem falhar”.

As atividades de algumas dessas organizações se tornam inviáveis se os serviços prestados pela rede não estiverem disponíveis, ou se forem prestados com tempos de resposta acima de determinados limites. À medida que as redes locais crescem e se interligam com redes de outras organizações, torna-se necessária a utilização de ferramentas que facilitem sua gerência. (ALBUQUERQUE, 2001).

Hoje, o SNMP (*Simple Network Management Protocol*) é o protocolo mais utilizado para controlar redes comerciais de diversos tipos. O SNMP é um protocolo relativamente simples, contudo seu poder de gerenciamento é bastante poderoso, podendo controlar difíceis problemas apresentados em variados tipos de redes TCP/IP.

Nesse contexto, existem ferramentas para o gerenciamento de rede baseadas no protocolo SNMP, com a vantagem adicional de serem distribuídas gratuitamente com versões disponíveis para os mais variados sistemas operacionais.

Essa integração de ferramentas livres baseadas no protocolo SNMP para gerência de redes pode se tornar, portanto, um vantajoso meio para as organizações conseguirem realizar um eficiente controle sobre suas redes de computadores utilizando os recursos oferecidos pelas ferramentas livres.

Faz-se inevitável realizar análises referentes aos recursos de cada *software* a fim de se identificar vantagens, desvantagens e a viabilidade para implementação em organizações que necessitem de uma correta gerência das informações em suas redes de computadores.

4 REVISÃO DE LITERATURA

Neste capítulo do trabalho são abordados todos os temas e explicações de cada conceito apresentado no mesmo. Faz-se necessária a leitura e entendimento de cada tópico para um completo conhecimento do assunto tratado neste documento.

4.1 GERENCIAMENTO DE REDES

De acordo com Black (2008), o gerenciamento de redes é uma atividade importante para manter as mesmas operando corretamente. Para se realizar tais tarefas gerenciais, o uso de *softwares* específicos (aqui também chamados de ferramentas) tornou-se uma constante, dado o notório aumento do número de dispositivos a ser gerenciado, o que impede um tratamento individualizado de cada um, bem como dado à necessidade de procedimentos automatizados de configuração, monitoração, reportes, entre outros.

As redes prestam serviços fundamentais na maioria das organizações. As atividades de algumas dessas organizações se tornam inviáveis se os serviços prestados pela rede não estiverem disponíveis, ou se forem prestados com tempos de resposta acima de determinados limites. À medida que as redes locais crescem e se interligam com redes de outras organizações, torna-se necessária a utilização de sistemas que facilitem sua gerência (ALBUQUERQUE, 2001).

A gerência está associada ao controle de atividades e ao monitoramento do uso de recursos da rede. As tarefas básicas de gerência em redes são: obter informações da rede, tratar estas informações possibilitando um diagnóstico e encaminhar as soluções dos problemas (SZTAJNBERG, 1996).

Nessa mesma linha de pensamento, conforme Delfino (1998), os objetivos do gerenciamento são:

- Identificação e registro de problemas (Até antes de o usuário perceber);
- Determinação da causa (Quem são os culpados?);
- Registrar a ocorrência de eventos (Possíveis problemas?);
- Prevenir a ocorrência de falhas (Correlação de eventos);
- Controlar os recursos da rede. Administrar a configuração da rede;

- Monitorar o desempenho da rede (Planejar seu crescimento);
- Gerenciar a segurança da rede;
- Definir pontos, elementos e parâmetros críticos para alarmes.

Stange (2008) diz que além dos sistemas de gerenciamento é fundamental que o responsável por uma rede tenha amplos conhecimentos de procedimentos, desempenho e identificação de falhas que possam acontecer. Outra característica essencial ao administrador ou gerente de uma rede é a familiarização com os sistemas por ele utilizados no cotidiano.

Os sistemas usados na gerência de redes procuram prestar os serviços sem sobrecarregar as entidades gerenciadas ou canais de comunicação e de forma objetiva.

Segundo Tanenbaum (2003), os componentes de um sistema de gerenciamento são:

a) dispositivos gerenciados: são dispositivos de *hardware*, como os computadores, roteadores e serviços de terminais, que estão conectados à rede;

b) agentes: são programas que residem nos elementos da rede que devem ser gerenciados. Eles coletam e armazenam diversas informações de gerenciamento;

c) base de informação de gerenciamento (*Management Information Base – MIB*): é uma estrutura de dados que contém uma relação dos objetos gerenciáveis. Os dados contidos nesta estrutura são obtidos pelos agentes e armazenados nesta estrutura;

d) gerentes: são *softwares* que concentram os dados obtidos sobre os diversos dispositivos da rede e os disponibilizam já interpretados para o gerente da rede;

e) protocolos de gerenciamento: através destes protocolos é possível estabelecer a interação entre os programas gerentes e agentes;

f) interfaces gráficas com o usuário (*Graphical User Interface – GUI*): nelas a aplicação disponibiliza de forma amigável os dados e as informações para o usuário.

4.1.1 Importância do gerenciamento de redes

Segundo Stallings (1999), o gerenciamento e monitoração de redes são tarefas extremamente importantes para a saúde de uma rede de computadores, sendo que, sem operações de gerenciamento, uma rede local não tem como manter-se operacional por muito tempo. Em especial, grandes redes corporativas estão fadadas ao caos sem estas funções. Além de agirem reativamente, as tarefas gerenciais de rede também são proativas no sentido de prevenir e detectar possíveis problemas.

Segundo Martin-Flatin, Znaty e Hubaux (1999), uma aplicação de gerenciamento é composta por gerentes executando nas estações de gerenciamento e agentes executando nos elementos gerenciados. O termo gerente pode ser utilizado, também, para designar a pessoa responsável pelo gerenciamento da rede e, sendo assim, para evitar problemas de interpretação, serão utilizados os termos, operador e administrador, nestes casos, ficando o termo gerente exclusivo para denominar as entidades de *software*.

Gerenciar uma rede é uma atividade bastante trabalhosa. Nos últimos anos o tráfego de informações dentro das redes corporativas aumentou exponencialmente devido ao surgimento de inúmeras novas aplicações. Concorrentemente, novas tecnologias e padrões proporcionaram uma grande proliferação de dispositivos heterogêneos conectados à rede.

A área de gerência de redes foi inicialmente impulsionada pela necessidade de monitoração e controle do universo de dispositivos que compõem as redes de comunicação. Com esta crescente necessidade de gerenciamento, fez-se necessário que padrões para ferramentas fossem estabelecidos.

Em resposta a esta necessidade, como relata Black (2008), surgiram dois padrões:

- Família de Protocolos SNMP: o protocolo *Simple Network Management Protocol* (SNMP) refere-se a um conjunto de padrões para gerenciamento que inclui um protocolo, uma especificação de estrutura de dados, e um conjunto de objetos de dados. Este é o protocolo de gerência adotado como padrão para redes TCP/IP.

- Sistemas de gerenciamento OSI: este termo refere-se a um grande conjunto de padrões de grande complexidade, que definem aplicações de propósito gerais

para gerência de redes, um serviço de gerenciamento e protocolo, uma especificação de estrutura de dados, e um conjunto de objetos de dados.

Este conjunto de protocolos é conhecido como *Common Management Information Protocol* (CMIP), mas, pela sua complexidade e lentidão do processo de padronização, este sistema de gerenciamento não é muito popular. (STALLINGS, 1999).

O gerenciamento da rede realizado pelo protocolo SNMP, permite que uma ou mais máquinas na rede sejam designadas gerentes da rede. Estas máquinas recebem informações de todas as outras máquinas da rede, chamadas agentes, e através do processamento destas informações pode gerenciar toda a rede e detectar facilmente problemas ocorridos. As informações coletadas pela máquina gerente estão armazenadas nas próprias máquinas da rede, em uma base de dados conhecida como *Management Information Base* (MIB). Nesta base de dados estão gravadas todas as informações necessárias para o gerenciamento deste dispositivo, através de variáveis que são requeridas pela estação gerente. Entretanto, em uma interligação de diversas redes locais, pode ser que uma rede local esteja funcionando perfeitamente, mas sem conexão com as outras redes, e, conseqüentemente, sem conexão com a máquina gerente. O ideal é implementar em alguma máquina, dentro desta rede local, um protocolo para gerenciamento que permita um trabalho *off-line*, isto é, que a rede local possa ser gerenciada, ou pelo menos tenha suas informações de gerenciamento coletadas, mesmo que estas informações não sejam enviadas instantaneamente a estação gerente. (BLACK, 2008).

Black (2008), a título de curiosidade, diz que o protocolo *Remote Monitoring* (RMON),

[...] permite uma implementação neste sentido ilustrado acima, devendo ser implementado em diversas máquinas ao longo da rede. É possível, ainda, que uma estação com implementação RMON, envie dados à estação gerente apenas em uma situação de falha na rede. Isto contribuiria para redução do tráfego de informações de controle na rede (*overhead*), facilitando seu gerenciamento, propiciando-se a instalação de um servidor *proxy*, que, além de servir como *cache* dos documentos acessados por uma rede local, pode também restringir o acesso a alguns documentos ou a utilização de algum protocolo, garantindo segurança à rede.

4.1.2 Necessidade do gerenciamento de redes

Por menor e mais simples que seja uma rede de computadores, ela precisa ser gerenciada, a fim de garantir, aos seus usuários, a disponibilidade de serviços a um nível de desempenho aceitável. À medida que a rede cresce, aumenta a complexidade de seu gerenciamento, forçando a adoção de ferramentas automatizadas para a sua monitoração e controle. A adoção de um *software* de gerenciamento não resolve todos os problemas da pessoa responsável pela administração da rede. Geralmente o usuário de um *software* de gerenciamento espera muito dele e, conseqüentemente, fica frustrado quanto aos resultados que obtém. Por outro lado, esse mesmo *software* quase sempre é subutilizado, isto é, possui inúmeras características inexploradas ou utilizadas de modo pouco eficiente. Para gerenciar um recurso, é necessário conhecê-lo muito bem e visualizar claramente o que este recurso representa no contexto da rede. (SANTOS, 2005).

De acordo com Black (2008), o investimento em um *software* de gerenciamento pode ser justificado pelos seguintes fatores:

- As redes e recursos de computação distribuídos estão se tornando vitais para a maioria das organizações. Sem um controle efetivo, os recursos não proporcionam o retorno que a corporação requer.

- O contínuo crescimento da rede em termos de componentes, usuários, *interfaces*, protocolos e fornecedores ameaçam o gerenciamento com perda de controle sobre o que está conectado na rede e como os recursos estão sendo utilizados.

- Os usuários esperam uma melhoria dos serviços oferecidos (ou no mínimo, a mesma qualidade), quando novos recursos são adicionados ou quando são distribuídos.

- Os recursos computacionais e as informações da organização geram vários grupos de aplicações de usuários com diferentes necessidades de suporte nas áreas de desempenho, disponibilidade e segurança. O gerente da rede deve atribuir e controlar recursos para balancear estas várias necessidades.

- À medida que um recurso fica mais importante para a organização, maior fica a sua necessidade de disponibilidade. O sistema de gerenciamento deve garantir esta disponibilidade.

- A utilização dos recursos deve ser monitorada e controlada para garantir que as necessidades dos usuários sejam satisfeitas a um custo razoável. Além desta visão qualitativa, uma separação funcional de necessidades no processo de gerenciamento foi apresentada pela *International Organization for Standardization* (ISO), como parte de sua especificação de Gerenciamento de Sistemas OSI. Esta divisão funcional foi adotada pela maioria dos fornecedores de sistemas de gerenciamento de redes para descrever as necessidades de gerenciamento: Falhas, Desempenho, Configuração, Contabilização e Segurança.

4.1.3 Áreas da gerência

A gerência de rede possui cinco áreas que segundo Castaldin (2005), em ordem de importância são mostradas a seguir:

4.1.4 Gerência de configuração

O alvo da gerência de configuração é o de aceitar a elaboração, a introdução, a partida, a operação contínua, e a posterior suspensão dos serviços de interconexão entre os sistemas abertos, tendo então, o emprego de manutenção e monitoração da estrutura física e lógica de uma rede, abrangendo a averiguação da existência dos elementos, e a verificação da interconectividade entre estes elementos.

A gerência de configuração, logo, é correspondente a um conjunto de facilidades que permitem controlar os objetos gerenciados, identificá-los, coletar e disponibilizar dados sobre estes objetos para as funções de atribuir valores iniciais e fazer alterações aos parâmetros de um sistema aberto e iniciar e encerrar as operações dos objetos gerenciados.

4.1.5 Gerência de falhas

A gerência de falhas é responsável pela detecção, isolamento e conserto de falhas na rede. As informações que são coletadas sobre os vários recursos da rede podem ser usadas em conjunto com um mapa desta rede, para indicar quais

elementos estão funcionando, quais estão em mau funcionamento, e quais não estão funcionando.

O ideal é que as falhas que possam vir a ocorrer em um sistema sejam detectadas antes que os efeitos significativos decorrentes desta falha sejam percebidos.

4.1.6 Gerência de desempenho

A gerência de desempenho faz o papel da monitoração de desempenho, da análise desse desempenho e planejamento de capacidade da rede.

O gerenciamento de desempenho é um conjunto de funções responsáveis pela manutenção e exame dos registros que contém o histórico dos estados de um sistema, com o objetivo de serem usados na análise das tendências do uso dos componentes, e para definir um planejamento do sistema através do dimensionamento dos recursos que devem ser alocados para o sistema, com o objetivo de atender aos requisitos dos usuários deste sistema, para satisfazer a demanda de seus usuários, ou seja, garantir que não ocorram insuficiências de recursos quando sua utilização se aproximar da capacidade total do sistema.

4.1.7 Gerência de segurança

Na gerência de segurança, a atenção está voltada pela proteção dos elementos da rede, monitorando e detectando violações da política de segurança estabelecida.

O objetivo do gerenciamento de segurança é o de dar subsídios à aplicação de políticas de segurança, que são os aspectos essenciais para que uma rede seja operada corretamente, protegendo os objetos gerenciados e o sistema de acessos indevidos de intrusos.

Deve providenciar um alarme ao gerente da rede sempre que se detectarem eventos relativos à segurança do sistema. Os mecanismos a serem adotados dependem do uso de uma política de segurança, que é feita pelo uso das funções de segurança do gerenciamento de redes.

4.1.8 Gerência de contabilidade

Responsável pela contabilização e verificação de limites da utilização de recursos da rede, com a divisão de contas feita por usuários ou grupos de usuários.

A gerência de contabilidade provê meios para se medir e coletar informações a respeito da utilização dos recursos e serviços de uma rede, para podermos saber qual a taxa de uso destes recursos, para garantir que os dados estejam sempre disponíveis quando for necessário ao sistema de gerenciamento, ou durante a fase de coleta, ou em qualquer outra fase posterior a esta. Deve existir um padrão para obtenção e para a representação das informações de contabilização, e para permitir a interoperabilidade entre os serviços da rede.

4.1.9 Histórico do gerenciamento de redes

Quando em 1986 reuniu-se, pela primeira vez, o Grupo de Trabalho sobre gerenciamento de Redes do Comitê Técnico em Comunicação de dados *International Federation for Information Processing* (IFIP) havia apenas o consenso sobre a necessidade de gerenciamento. Cerca de 20 pessoas reunidas em Dallas, provenientes de diversos países, sequer concordavam sobre o escopo do gerenciamento de rede. Enquanto representantes incorporassem apenas as três camadas inferiores da arquitetura *Open Systems Interconnection* (OSI) (pois era como se estava acostumado a trabalhar), para os outros o gerenciamento de redes devia englobar as sete camadas. Percebia-se claramente que cada fornecedor tinha construído uma arquitetura proprietária de gerenciamento para seus produtos e tinha dificuldade de impingi-la aos clientes, ao lado de outros fornecedores. Já se falava na oportunidade sobre o gerenciamento OSI, embora muitos tenham encarado com certo ar de dúvida aquela alternativa. (BLACK, 2008).

Santos (2005), nessa mesma linha, relata que a abordagem clássica para integrar o gerenciamento de redes era, pois, baseada em arquitetura proprietária. Para que pudessem funcionar como elemento de integração, os arquitetos de tais soluções incorporaram nelas uma abertura para agregar a informação e gerenciamento de sistema de outros fornecedores. A IBM, por exemplo, com o conceito de ponto focal abriu esta porta para integrar outros sistemas de

gerenciamento ao *Netview*, principalmente por interesse próprio, uma vez que a aquisição da RDLM (fabricante PABX) levou a esta necessidade. Módulos para traduzir o fluxo de informação de gerenciamento de um esquema para outros tinham de ser constituídos e podiam ser implantados em vários pontos, como mostra a Figura 1:

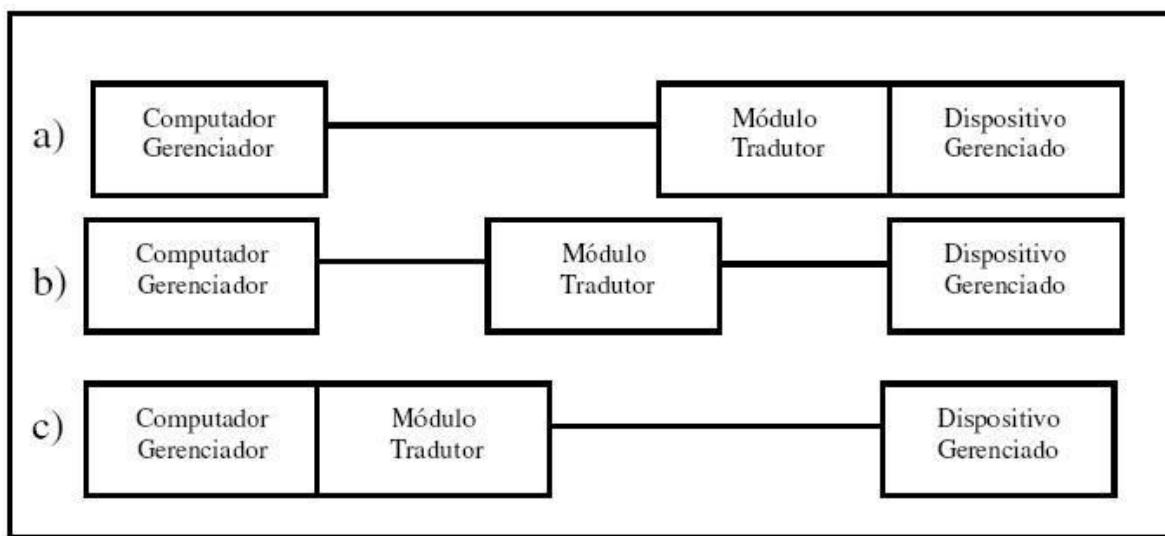


Figura 1 - Formas de Implantação do Módulo Tradutor.
Fonte: Santos (2005, p. 17).

Santos (2005) mostra que na Figura 1, a) poderia ser um servidor de rede *Novell*, com um módulo interno capaz de gerar os “vetores de alerta” esperados pelo *Netview* e b) poderia ser a solução para integrar o gerenciamento de um PABX digital em que a tradução seria feita em um PC que receberia as mensagens de gerenciamento de um lado e as traduziria, quando possível, para o outro. A terceira abordagem seria para o caso em que um roteador fosse o dispositivo gerenciado e que usasse um protocolo padrão de fato na indústria, tal como o *SNMP* (*Simple Network Management Protocol* da arquitetura Internet), com a conversão feita internamente no computador gerenciador.

Santos (2005) prossegue e diz que dentro dos problemas decorrentes desta solução, pode-se destacar principalmente a limitação imposta pelo fato de somente usar opções gerenciamento (dados recebidos e comandos veiculáveis) que tinham similar na arquitetura proprietária do fornecedor do computador gerenciador. Opções de interação propiciadas pelos dispositivos gerenciados podiam não ser aproveitadas simplesmente pela falta de condições de mapeá-las para uma forma

passível de reconhecimento pelo computador gerenciador. Em decorrência, os dispositivos gerenciados providos pelo mesmo fornecedor do computador gerenciador apareciam mais facilmente.

Para não parecerem diminuídos sob este prisma, muitos fornecedores não se mostravam entusiasmados em cooperar para tornar seus produtos gerenciáveis por um computador gerenciador de outro fabricante. Esta abordagem foi adotada por alguns fornecedores no mercado, como a IBM e a DEC, mas cada vez mais acrescida do desejo de que um sistema de gerenciamento independente de fornecedor pudesse rodar numa máquina dedicada, de modo a não sobrecarregar nem prejudicar o atendimento dos serviços normais a serem executados no mainframe. A AT&T também entrou no cenário, definindo uma arquitetura de gerenciamento e se propondo a gerenciar as redes de seus clientes de telecomunicações. Criando o impasse, uma solução alternativa teria de ser buscada, implicando a agregação de esforços que levassem a uma solução mais universal e padronizada. Obviamente, tal solução deveria englobar os serviços de gerenciamento mais importantes e relevantes, além de formalizar a interação entre os dispositivos gerenciados e os gerenciadores. A ISO tomou a bandeira e o esquema básico da arquitetura de gerenciamento de rede foi adicionado ao modelo de referência ISO/OSI em 1989. (BLACK, 2008).

A colaboração entre a ISO/*International Electrotechnical Committee* (IEC) resultou na série de documentos X.700, cujo objetivo maior foi criar condições para o desenvolvimento de produtos de gerenciamento de redes de computadores e sistema de comunicações heterogêneos.

Todavia, o embate das forças dominantes no cenário internacional dificultou a estabilização dos detalhes operacionais do modelo de gerenciamento. Anos se passaram sem que os documentos atingissem o estágio do padrão ISO internacional. As implantações, baseadas em interpretações da documentação disponível, começaram a aparecer e, em 1989, percebendo a necessidade de acordos que assegurassem a interoperabilidade das implementações, os fornecedores começaram a reunir-se em associações como a ISO/NM Fórum, para buscar um acordo que viabilizasse a definição de um conjunto de opções de implantação capaz de assegurar a interoperabilidade dos sistemas de gerenciamento. Outro grupo foi criado sob a tutela do *National Institute of Standards and Technology* (NIST) dos Estados Unidos para atender às necessidades do

governo americano, que já havia determinado, através de seu documento *Government OSI Profile* (GOSIP), que as soluções de redes a serem adquiridas deveriam atender às recomendações ISO/IEC. Este trabalho resultou no *Government Network Management Profile* (GNMP), cuja versão 1, de 30 de julho de 1992, constitui a referência que todas as agências do governo federal dos Estados Unidos devem usar ao adquirir funções e serviços de gerenciamento de rede. (SANTOS, 2005).

Black (2008) conta que, o primeiro dos protocolos de gerência de rede foi o *Simple Gateway Monitoring Protocol* (SGMP) que surgiu em novembro 1987. Entretanto, o SGMP era restrito à monitoração de *gateways*. A necessidade crescente de uma ferramenta de gerenciamento de rede mais genérica fez emergir mais algumas abordagens:

- *High-Level Entity Management System* (HEMS) – generalização do *Host Management Protocol* (HMP);

- *Simple Network Management Protocol* (SNMP) – um melhoramento do SGMP;

- CMIP over TCP/IP (CMOT) uma tentativa de incorporar o máximo possível o protocolo (CMIP), serviços e estrutura de base de dados que estava sendo padronizada pela ISO para gerenciamento de redes.

No início de 1988 a *Internet Architecture Board* (IAB) revisou os protocolos e escolheu o SNMP como uma solução de curto prazo e o CMOT como solução de longo prazo para o gerenciamento de redes. O sentimento era que, em um período de tempo razoável, as instalações migrariam do TCP/IP para protocolos baseados em OSI. Entretanto, como a padronização do gerenciamento baseado no modelo OSI apresentava muita complexidade de implementação e o SNMP, devido à sua simplicidade, foi amplamente implementado nos produtos comerciais, o SNMP tornou-se um padrão de fato. Posteriormente, pela existência de lacunas funcionais (devido exatamente à simplicidade do SNMP), foram definidas novas versões do protocolo SNMP chamadas de SNMPv2 e SNMPv3, e o SNMP original ficou conhecido como SNMPv1. (BLACK, 2008).

Assim, Black (2008) finaliza dizendo que a primeira versão da arquitetura de gerenciamento SNMP foi definida no RFC 1157 de maio de 1990. O RFC 1157 define ainda três objetivos a serem alcançados pelo SNMP: minimizar o número e complexidade das funções de gerenciamento, ser flexível o suficiente para permitir

expansões futuras e ser independente da arquitetura e mecanismo dos dispositivos gerenciados. A definição das informações de gerenciamento requer não apenas profundo conhecimento da área específica em foco, mas também do modelo de gerenciamento.

4.2 PROTOCOLO SNMP (SIMPLE NETWORK MANAGEMENT PROTOCOL)

Como explica Harnedy (1997), o *Simple Network Management Protocol* (SNMP) é um protocolo da camada de aplicação lançado em 1988 e projetado para facilitar a troca de informação de gerência entre dispositivos da rede. Através do SNMP, são transportados dados informativos (tais como pacotes por taxas de segundo e de erro da rede).

Utiliza os serviços do protocolo de transporte UDP (*User Datagram Protocol*) para enviar suas mensagens através da rede. Com esse protocolo, os administradores controlam facilmente o desempenho da rede, encontram e resolvem problemas. Sua especificação está contida no RFC-1155 (*Structure Of Management Information*), RFC-1156 (*Management Information Base*) e RFC-1157 (*Simple Network Management Protocol*). Este protocolo é o centro do desenvolvimento do gerenciamento SNMP. (MELO, 2007).

Como o *Transmission Control Protocol* (TCP), o SNMP é um *Internet Protocol*. Atualmente, existem três versões do SNMP: versão 1, versão 2 e versão 3. Na versão 2 do SNMP, procurou-se a correção de algumas deficiências da versão 1, melhorando a comunicação através da chamada *Manager to Manager* MIB. Já a versão 3 do SNMP tem como vantagens aspectos ligados à segurança.

Conforme Melo (2007), hoje, o SNMP é o protocolo mais utilizado para controlar redes comerciais de diversos tipos. O SNMP é um protocolo relativamente simples, contudo seu poder de gerenciamento é bastante poderoso, podendo controlar difíceis problemas apresentados em variados tipos de redes TCP/IP.

O SNMP, segundo Harnedy (1997), parte do esquema de gerenciamento OSI, onde os processos que implementam as funções de gerenciamento de Internet atuam como agentes ou gerentes. Esses agentes têm por função descobrir falhas ou problemas nos componentes da rede (*Hosts*, roteadores, *gateways*, etc...). Dessa forma, podem ser tomadas providências antes mesmo que o problema venha a acontecer, ou até mesmo saber como ou de onde surgiu o problema.

Cada componente gerenciado é visto como uma coleção de variáveis, onde os valores podem ser lidos ou alterados. O gerente, então, envia comandos aos agentes, solicitando uma leitura no valor das variáveis dos componentes gerenciados, ou modificando seu valor. Na troca de informações entre o gerente e o agente, são aplicados mecanismos de autenticação para evitar que usuários não autorizados interfiram no funcionamento da rede. Essa troca de mensagem entre gerente e agente é definida pelo protocolo SNMP, onde ele define o formato e a ordem que deve ser seguida à seqüência das informações de gerenciamento.

Para armazenar tais informações, são utilizados MIB (*Management Information Base*), onde são armazenadas as informações sobre o funcionamento dos *Hosts*, Roteadores e dos processos que executem os protocolos de comunicação (TCP, IP, ARP, etc...). Com o SNMP os gerentes de rede têm também a capacidade de modificar valores de uma variável de um objeto na MIB. (MELO, 2007).

4.2.1 Principais objetivos do protocolo SNMP

De acordo com Black (2008), os principais objetivos do protocolo SNMP são:

- Reduzir o custo da construção de um agente que suporte o protocolo;
- Reduzir o tráfego de mensagens de gerenciamento pela rede;
- Reduzir o número de restrições impostas às ferramentas de gerenciamento da rede, devido ao uso de operações complexas e pouco flexíveis;
- Apresentar operações simples de serem entendidas, sendo facilmente usadas pelos desenvolvedores de ferramentas de gerenciamento;
- Permitir facilmente a introdução de novas características e novos objetos não previstos ao se definir o protocolo;
- Construir uma arquitetura que seja independente de detalhes relevantes e priorize somente algumas implementações particulares.

4.2.2 Agente e gerente

Comer (1999, p. 437) cita que,

Agente é um processo executado em uma máquina gerenciada, sendo responsável pela manutenção das informações de gerência da máquina. Ele tem duas funções principais: atender as requisições enviadas pelo gerente e

enviar automaticamente informações de gerenciamento ao gerente quando previamente programado.

Comer (1999, p. 437) diz também que,

Gerente é um programa executado trabalhando em uma estação servidora, permitindo a obtenção e envio de informações de gerenciamento junto aos dispositivos gerenciados mediante a comunicação com um ou mais agentes. Ele é responsável pelo monitoramento, relatórios e decisões na ocorrência de problemas, enquanto que o agente fica responsável pelas funções de envio e alteração das informações e também pela notificação da ocorrência de eventos específicos ao gerente.

“Resumindo, a gerência de redes que utiliza o protocolo SNMP consiste em quatro componentes principais: nós gerenciados, estações de gerenciamento, informações de gerenciamento e um protocolo de gerenciamento”. (SOARES, 1995, p. 419).

A Figura 2 ilustra um sistema envolvendo a troca de informações entre Agente e Gerente:

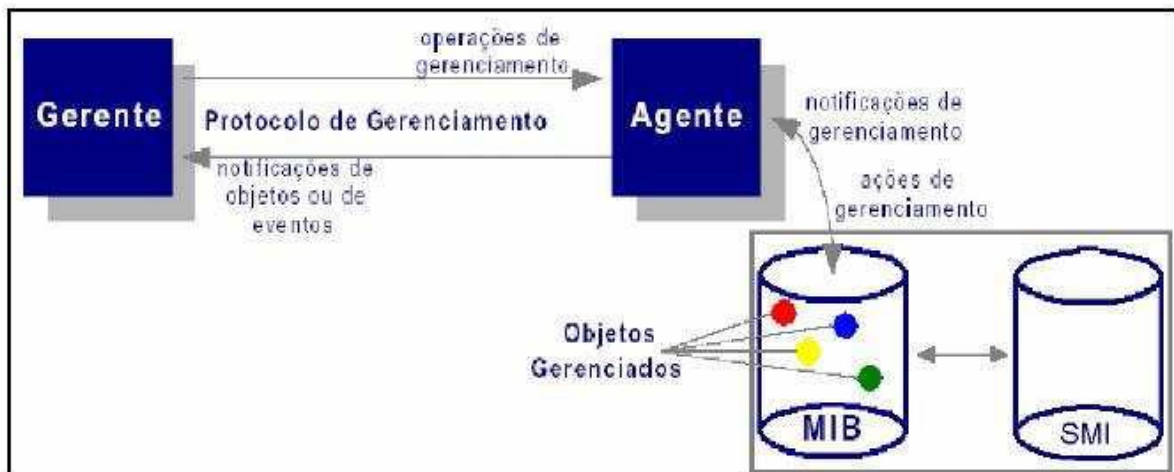


Figura 2 - Componentes do SNMP
Fonte: Harnedy (1997)

4.2.3 MIB – *Management Information Base*

Para Harnedy (1997), as MIBs, ou bases de informações de gerência, são compostas pelas informações de gerenciamento e pelos objetos gerenciados. Um objeto gerenciado é definido como a unidade da informação de gerenciamento. A comunicação e o processamento de dados são os recursos que podem ser gerenciáveis através da utilização de um protocolo. As informações de

gerenciamento referentes aos objetos gerenciados residem na MIB. Ela define o conteúdo da informação que é transportada através do protocolo de gerenciamento.

Tradicionalmente, define-se uma MIB como um conjunto de objetos gerenciados. Estes objetos e suas instâncias são representados por variáveis. Às variáveis são atribuídas definições que informam exatamente quais serão seus atributos.

A *Management Information Base* do Agente é uma coleção de variáveis de interesse. Os grupos de variáveis da MIB que compreendem um particular módulo de informação de gerenciamento controlado pelo Agente são dependentes das funcionalidades do dispositivo e de quais recursos ou serviços o Agente deverá gerenciar. (HARNEDY, 1997).

4.2.4 Estrutura da MIB

A estrutura da informação de gerenciamento (*SMI – Structure of Management Information*) define as regras para a descrição da informação de gerenciamento. (SOARES, 1995).

O SMI atende às necessidades para que as variáveis que representam os objetos gerenciados da rede sejam adequadamente definidas. Sua estrutura é em forma de árvore, cuja função primária é definir os nomes das variáveis da MIB. Segundo Comer (1999), cada objeto ao qual o SNMP tem acesso deve ser definido e determinado com um único nome. O SMI é definido utilizando a linguagem ASN.1 (*Abstract Syntax Notation 1*), permitindo que a MIB possa ser definida e categorizada de acordo com a estrutura hierárquica definida.

O ASN.1 atribui a cada objeto um prefixo longo que garante que o nome será único. Por exemplo, um inteiro que conta o número de datagramas IP que um dispositivo recebe tem o nome: *iso.org.dod.internet.mgmt.mib.ip.ipInReceives*. Quando o nome do objeto for representado em uma mensagem SNMP, em cada parte do nome é atribuído um inteiro. Deste modo cada variável tem sua identificação única ou OID (*ObjectID*). Por exemplo, em uma mensagem SNMP o nome *ipInReceives* é: 1.3.6.1.2.1.4.3. (MELO, 2007).

A MIB foi definida primeiramente em oito grupos de objetos, logo depois, com o lançamento da MIB II, foram incorporados mais dois grupos. Os grupos que

constituem a MIB são: *System*, *Interfaces*, *Address translation*, IP, ICMP, TCP, UDP, EGP, *Transmission*, SNMP. Cada um destes grupos define operações e novos grupos. Por exemplo: o grupo *System* descreve o *hardware* e o sistema operacional da máquina gerenciada. (MELO, 2007).

A estrutura de árvore da MIB pode ser visualizada na Figura 3.

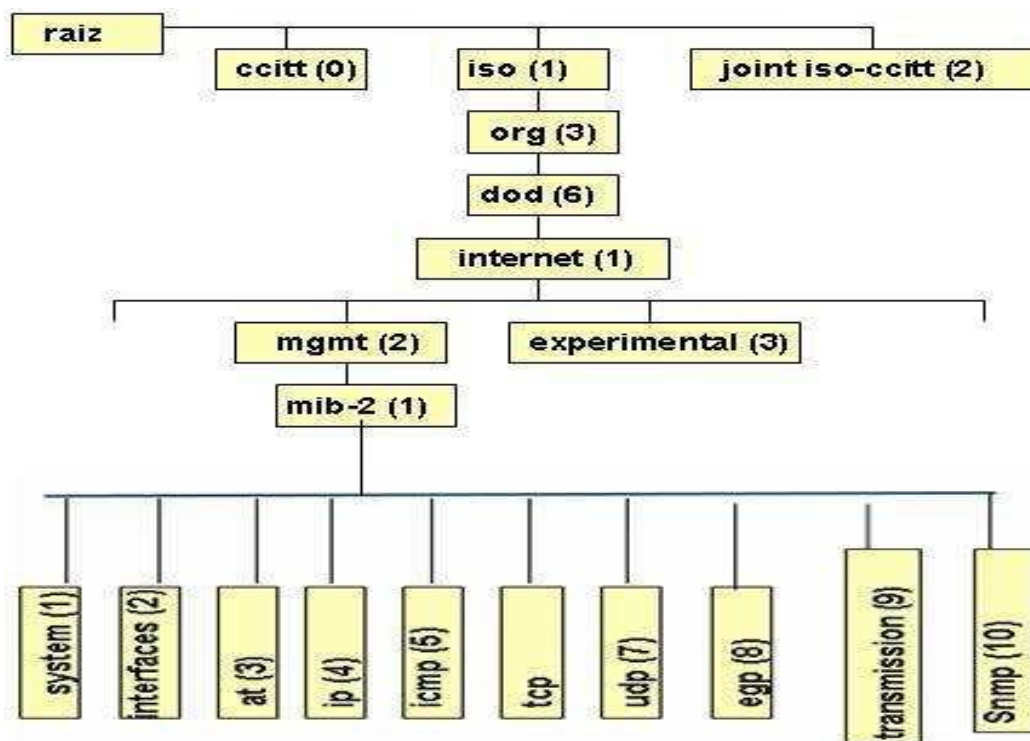


Figura 3 - Estrutura da MIB
Fonte: Soares (1995)

4.2.5 Pontos positivos e negativos do snmp

O SNMP tem vários pontos positivos. Um deles é sua popularidade para a gerência de redes TCP/IP. Agentes SNMP estão disponíveis para vários dispositivos de rede, desde computadores até *bridges*, *modems* ou impressoras (STALLINGS, 1999).

Adicionalmente, o SNMP é um protocolo de gerenciamento flexível e extensível. Pode-se estender os agentes SNMP para cobrir dados específicos de dispositivos, pelo uso de arquivos ASN.1. O SNMP pode assumir numerosos trabalhos específicos para classes de dispositivos fornecendo um mecanismo padrão de controle de rede e monitoramento (MELLO, 2000).

Quanto aos pontos negativos do SNMP, estes podem ser descritos principalmente pela utilização de grandes pacotes para pequenas informações e falhas de segurança. A utilização de pacotes de tamanho excessivo ocorre principalmente pela forma como são identificados os objetos de gerenciamento. Estes objetos recebem nomenclaturas em forma de uma seqüência de *bit*, onde cada *bit* representa uma especificação da MIB. Dessa forma existe um tráfego desnecessário de informações na rede. (STANGE, 2008).

Outra deficiência do SNMP padrão está nas brechas de segurança que podem permitir o acesso de intrusos às informações transportadas pela rede. Esses intrusos podem, portanto, acessando estas informações, retirar algumas máquinas da rede. A solução para este problema, no entanto, é simples: a expansão do SNMP. A versão SNMPv3, adiciona mecanismos de segurança que auxiliam no combate dos três maiores problemas de segurança: a privacidade dos dados (previne que intrusos tenham acesso às informações de gerenciamento transportadas pela rede), autenticação (previne que intrusos enviem dados falsos através da rede) e controle de acesso (restringe o acesso a determinadas variáveis para certos usuários, reduzindo a possibilidade de um usuário, acidentalmente, danificar a rede). (GOETEN, 2001).

O protocolo SNMP está longe da perfeição, contudo, devido a sua flexibilidade, os principais problemas relatados podem ser contornados e por isto é utilizado desde a década de 80 pelas grandes ou pequenas empresas fabricantes de equipamentos.

4.2.6 Operações do SNMP

Para obter informações de gerenciamento, o protocolo SNMP utiliza a troca de mensagens. Estas mensagens são compostas por um cabeçalho padrão e uma PDU (*Protocol Data Unit*). O cabeçalho especifica a versão do SNMP e o nome da comunidade. O primeiro serve para que a troca de mensagens entre agentes e gerentes seja compatível, o nome da comunidade serve como um dispositivo de segurança, ou uma senha de acesso. (MELO, 2007).

Ainda segundo Melo (2007), quando um agente recebe uma solicitação do gerente, imediatamente será feita uma solicitação da comunidade. Se a comunidade

for igual à definida pelo agente, o gerente terá o acesso, caso contrário, uma mensagem será retornada pelo agente iniciando falha na autenticação.

As PDUs definidas pelo SNMP, segundo Comer (1999), são as seguintes:

- *Get*: Utilizada para requisitar um ou mais valores da MIB do sistema;
- *Get-next*: Recupera os valores seqüencialmente;
- *Get-bulk* (SNMPv2 e SNMPv3);
- *Set*: Atualiza valores de variáveis;
- *Get-response*: Retorna os resultados das PDUs *get*, *get-next* e *set*;
- *Trap*: Informação do agente sobre eventos e problemas;
- *Notification* (SNMPv2 e SNMPv3);
- *Inform* (SNMPv2 e SNMPv3);
- *Report* (SNMPv2 e SNMPv3).

4.2.7 Funcionamento do snmp

O protocolo SNMP foi desenvolvido para rodar sobre a camada de transporte, na camada de aplicação da pilha de protocolo TCP/IP. A maioria das implementações do SNMP utilizam o *User Datagram Protocol* (UDP) como protocolo de transporte. O UDP é um protocolo não-confiável, não garantindo a entrega, a ordem ou a proteção contra duplicação das mensagens (GOETEN, 2001).

O SNMP utiliza o UDP, pois foi desenvolvido para funcionar sobre um serviço de transporte sem conexão. Foi adotada a utilização do UDP principalmente para não comprometer o desempenho da rede por onde trafegam as informações de gerenciamento. Como é exigido do serviço de gerenciamento, que este seja o mais rápido possível e sem comprometer desempenho, não seria eficiente utilizar um protocolo que dependesse de um serviço orientado a conexão ou que necessitasse de confirmações a cada mensagem. Estas confirmações gerariam um tráfego desnecessário na rede, comprometendo seu desempenho. (MELLO, 2000).

Como o UDP é um protocolo não-confiável, é possível que mensagens SNMP sejam perdidas. O SNMP por si só não fornece garantia sobre a entrega das mensagens. As ações a serem tomadas quando da perda de uma mensagem SNMP não são abordadas pelo padrão. No entanto, cada *software* de gerência aborda esta questão de maneira distinta. Existem casos onde o *software* ao fazer uma operação de requisição de valores e não consegue obter o valor, utiliza a falha para

determinar a indisponibilidade do equipamento e alertar o gerente. Outra ação tomada é repetir a requisição até que a mesma obtenha o resultado desejado. (MELLO, 2000).

O SNMP utiliza cinco comandos básicos para suas operações. O comando *Get-Request* solicita que os nomes das variáveis requeridos sejam explicitamente informados ao gerente. O comando *Get-Next-Request* solicita a variável seguinte, permitindo que um gerente percorra a MIB inteira alfabeticamente. O comando *Get-Bulk-Request* serve para a transferência de grandes quantidades de informação, como por exemplo, uma tabela de dados. A mensagem *Set-request* permite atualizar o valor de uma variável, mudando o estado desta, desde que a especificação do objeto permita essas atualizações. A mensagem *Inform-request* tem a utilidade de informar a um gerente quais as variáveis ele está gerenciando. O comando *Trap* é uma mensagem enviada de um agente para um gerente quando acionada. (STANGE, 2008)

A Figura 4 ilustra o contexto do protocolo SNMP na pilha de protocolo TCP/IP, utilizando o UDP como protocolo de transporte.

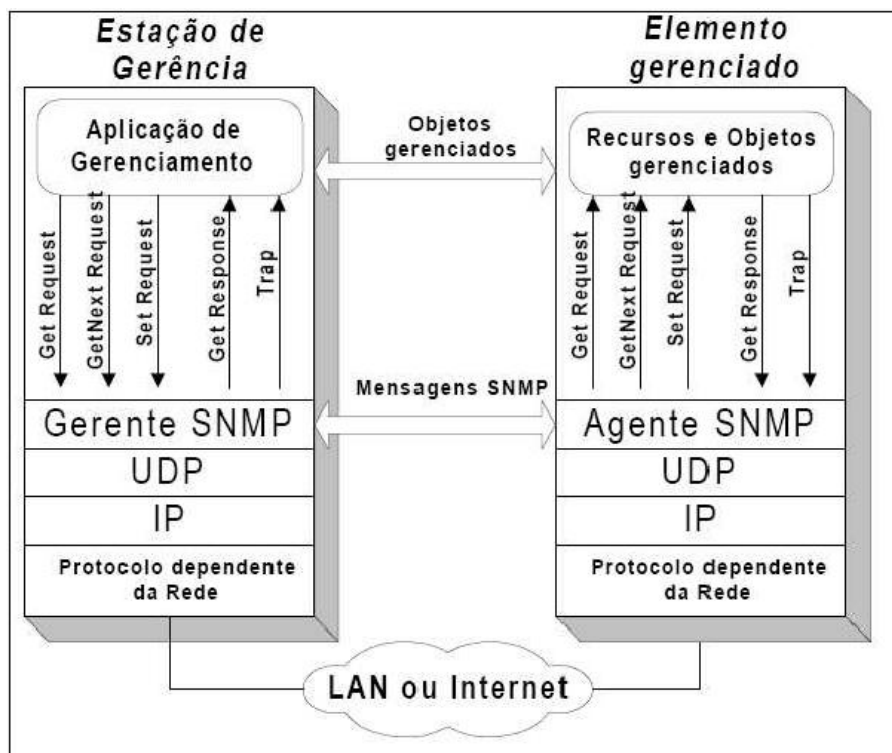


Figura 4 - Protocolo SNMP sobre a camada de transporte.
Fonte: Mello (2000).

As operações de requisição do SNMP como *Get*, *GetNext*, *GetBulk*, *Inform* e *Set* utilizam a porta 161, já operação *Trap* tem reservada para si a porta 162. Isto ocorre para que o tráfego seja separado e as informações possam ser transmitidas ou requeridas de forma mais segura. A formação das mensagens SNMP é feita, diferente da maioria dos protocolos, de forma inversa. Primeiramente é formado o pacote com as informações desejadas. Este pacote recebe então os indicadores de erros e requisições. Por fim o pacote formado recebe o cabeçalho de versão e comunidade. Tanto a versão como a comunidade devem ser as mesmas entre o gerente e o elemento gerenciado para que o pacote seja aceito e não descartado. (STANGE, 2008).

A Figura 5 ilustra o formato das mensagens SNMP através de pacotes.

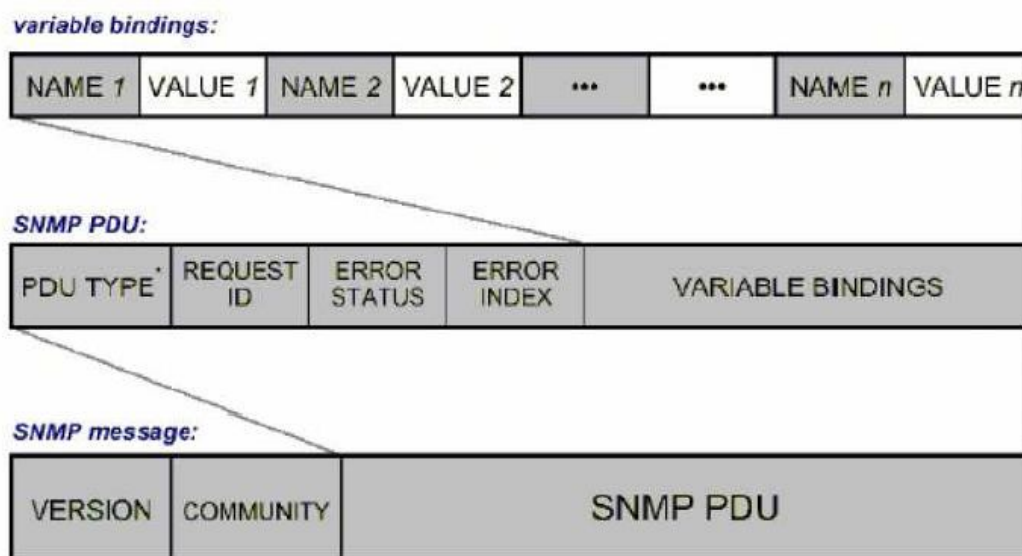


Figura 5 - Formato das mensagens SNMP.
Fonte: Schulz (2004).

4.3 SOFTWARE LIVRE

Segundo Campos (2006), *Software Livre*, ou *Free Software*, conforme a definição de *software livre* criada pela *Free Software Foundation*, é o *software* que pode ser usado, copiado, estudado, modificado e redistribuído sem restrição. A forma usual de um *software* ser distribuído livremente é sendo acompanhado por uma licença de *software livre* (como a GPL ou a BSD), e com a disponibilização do seu código-fonte.

Campos (2006) diz ainda que *software* Livre é diferente de *software* em domínio público. O primeiro, quando utilizado em combinação com licenças típicas (como as licenças GPL e BSD), garante os direitos autorais do programador/organização. O segundo caso acontece quando o autor do *software* renuncia à propriedade do programa (e todos os direitos associados) e este se torna bem comum.

O *Software* Livre como movimento organizado teve início em 1983, quando Richard Stallman deu início ao Projeto GNU e, posteriormente, à *Free Software Foundation*. *Software* Livre se refere à existência simultânea de quatro tipos de liberdade para os usuários do mesmo, definidas pela *Free Software Foundation*. (COSTA, 2010).

As 4 liberdades básicas associadas ao *software* livre são:

- A liberdade de executar o programa, para qualquer propósito (liberdade nº 0)
- A liberdade de estudar como o programa funciona, e adaptá-lo para as suas necessidades (liberdade nº 1). Acesso ao código-fonte é um pré-requisito para esta liberdade.
- A liberdade de redistribuir cópias de modo que você possa ajudar ao seu próximo (liberdade nº 2).
- A liberdade de aperfeiçoar o programa, e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie (liberdade nº 3). Acesso ao código-fonte é um pré-requisito para esta liberdade.

Um programa é *software* livre se os usuários têm todas essas liberdades. Portanto, o usuário deve ser livre para redistribuir cópias, seja com ou sem modificações, seja de graça ou cobrando uma taxa pela distribuição, para qualquer um em qualquer lugar. Ser livre para fazer essas coisas significa (entre outras coisas) que o usuário não tem que pedir ou pagar pela permissão, uma vez que esteja de posse do programa. (COSTA, 2010).

Deve-se também ter a liberdade de fazer modificações e usá-las privativamente no trabalho ou lazer, sem nem mesmo mencionar que elas existem. Se modificações forem publicadas, o usuário não deve ser obrigado a avisar a ninguém em particular, ou de nenhum modo em especial. (COSTA, 2010).

A liberdade de utilizar um programa significa a liberdade para qualquer tipo de pessoa física ou jurídica utilizar o *software* em qualquer tipo de sistema computacional, para qualquer tipo de trabalho ou atividade, sem que seja necessário

comunicar ao desenvolvedor ou a qualquer outra entidade em especial. (CAMPOS, 2006).

Costa (2010) faz questão de ressaltar que a liberdade de redistribuir cópias deve incluir formas binárias ou executáveis do programa, assim como o código-fonte, tanto para as versões originais quanto para as modificadas. De modo que a liberdade de fazer modificações, e de publicar versões aperfeiçoadas, tenha algum significado, deve-se ter acesso ao código-fonte do programa. Portanto, acesso ao código-fonte é uma condição necessária ao *software* livre.

Por fim, Campos (2006) lembra que, para que essas liberdades sejam reais, elas tem que ser irrevogáveis desde que o usuário não faça nada errado; caso o desenvolvedor do *software* tenha o poder de revogar a licença, mesmo que o usuário não tenha dado motivo, o *software* não é livre.

4.4 CACTI

De acordo com Black (2008), Cacti é uma ferramenta *freeware* que recolhe e exibe informações sobre o estado de uma rede de computadores através de gráficos, sendo um *frontend* para a ferramenta RRDTool, que armazena todos os dados necessários para criar gráficos e inseri-los em um banco de dados MySQL. Foi desenvolvida para ser flexível de modo a se adaptar facilmente a diversas necessidades, bem como ser robusta e fácil de usar. Monitora o estado de elementos de rede e programas bem como largura de banda utilizada e uso de CPU. O *frontend* foi escrito na linguagem PHP e contém suporte ao protocolo SNMP.

Costa (2008) diz que, RRDTool é um sistema de base de dados Round-Robin criado por Tobias Oetiker sob licença GNU/GPL. Foi desenvolvido para armazenar séries de dados numéricos sobre o estado de redes de computadores, porém pode ser empregado no armazenamento de qualquer outra série de dados como temperatura, uso de CPU, etc. RRD é um modo abreviado de se referir a Round Robin Database (base de dados *roundrobin*).

Com o Cacti é possível gerar gráficos referentes a uso de memória física, memória virtual, quantidade de processos, processamento, tráfego de rede, quantidade de espaço em disco, etc. Através do SNMP, permite ter acesso a gráfico não só de sistemas operacionais Linux, mas também de Windows e de dispositivos de rede como roteadores e *switches*, bem como qualquer dispositivo que suporte

SNMP. Todas as três versões do SNMP são suportadas atualmente pelo Cacti. (BLACK, 2008).

Sua arquitetura prevê a possibilidade de expansão através de inúmeros *plugins* desenvolvidos por sua comunidade que adicionam novas funcionalidades. Um bom exemplo destes *plugins* é o *PHP Network Weathermap* que mostra um mapa da rede e o estado de cada elemento. O produto permite aos usuários agendar serviços em intervalos pré-determinados gerando gráficos a partir destes resultados e ele permite lidar com diversos usuários simultâneos, cada um com seus gráficos gerados e com suas *queries* na rede, além de ser flexível, permitindo outros tipos de coletas de dados desde que obedeçam aos limites do RRDTool. (BLACK, 2008).

Costa (2008), também diz que o Cacti pode usar dois tipos de agentes remotos: o primeiro, um *script* PHP previsto para pequenas redes - via o arquivo *cmd.php*, ou então através do *poller "spine"* (antigamente chamado de agente ou *daemon cactid*), um pequeno agente escrito em C que pode ser amplamente escalado para grandes redes de computadores.

Uma vez instalado no sistema e logado, o administrador tem que informar o Cacti sobre os dispositivos que deseja controlar. Ele vem com uma lista de dispositivos comuns, tais como servidores Linux, roteadores Cisco, servidores *NetWare*, e até mesmo *workstations* Windows 2000/XP. Se o dispositivo não está na lista, você pode criar um dispositivo genérico e especificar os parâmetros que você precisa para monitorá-lo. Você também pode salvar isto como um modelo para uso futuro, sendo essa *interface web user-friendly*, junto com a documentação disponível, o destaque da ferramenta. (COSTA, 2008).

Depois de criar os dispositivos, é necessário selecionar os parâmetros que pretende acompanhar de cada dispositivo, e criar os gráficos. O Cacti fornece modelos para os parâmetros comuns, tais como o uso da CPU, o tráfego de rede, os usuários conectados e coisas do gênero, mas você pode rapidamente fazer seus próprios modelos também, bastando alguns minutos para criar os gráficos para servidores Linux/Windows. Os parâmetros para monitorar cargas médias de dispositivos são, por padrão, a largura de banda utilizada e os processos em execução, já oferecidos pela ferramenta. Para controlar os *switchers*/roteadores é mais complexo, mas a documentação é ampla e satisfatória. (COSTA, 2008)

As informações recolhidas são muitas e só serão úteis se apresentadas corretamente, sendo que, se há uma gama muito grande de dispositivos a serem monitorados, pode-se visualizar um pequeno número de gráficos, facilitando a administração do sistema, ao passo que dezenas ou centenas de parâmetros estão sendo monitorados, essa tarefa torna-se muito difícil. O Cacti permite que os gráficos gerados sejam organizados de diversos modos: configurando-os em forma de árvores ou agrupando todos os gráficos de um mesmo tipo sob um gráfico maior, podendo-se ter um gráfico em duas ou mais árvores também. Estas árvores de gráficos possuem diversas maneiras de serem organizadas, de acordo com a necessidade do administrador, podendo-se gerar gráficos de praticamente qualquer dispositivo que se deseje. A variedade de modelos que vêm com a instalação padrão é suficiente para cuidar de redes simples, e você pode criar seus próprios tipos de dados e modelos mais complexos para redes, apesar do Cacti não conseguir exibir e tabular dados numéricos. (BLACK, 2008).

Importante salientar que o Cacti não está limitado ao protocolo SNMP somente, pode-se alimentá-lo de outros modos – podendo apontá-lo para um caminho de um *script* ou comando externo – padrão **nix bash scripts, scripts Perl*, ou qualquer *script* que seja executado a partir do *prompt* de comando do servidores **nix*. O Cacti reúne os dados em uma tarefa *cron* e preenche uma base de dados MySQL própria armazenando os resultados.

Black (2008) lembra que nos *sites* de usuários de Cacti, há muitos *scripts* desenvolvidos para esses fins, que vão da coleta de dados em servidores Apache até filas de *e-mail* em servidores *Sendmail* para recolher estatísticas. O Cacti não exige demasiados recursos do *host* em que ele está rodando, pois foi escrito em PHP sobre plataforma *web*, sendo por natureza uma ferramenta ágil e rápida.

Pode-se autorizar vários administradores como usuários do Cacti ou então dando-lhes direitos restritos a apenas algumas áreas da ferramenta, permitindo criar usuários que podem alterar apenas alguns parâmetros de gráficos e outros que podem apenas visualizá-los, mas preservando as configurações individuais de cada um.

Por fim, Black (2008) ressalta que como pontos negativos destacam-se o fato do produto não possuir um agente de descoberta automático, ou seja, toda rede tem que ser adicionada manualmente, apesar de já haver *plugins* de terceiros que fazem esse trabalho – ainda assim, não é uma *feature* padrão da ferramenta, podendo

tornar o trabalho do administrador muito penoso se a rede for grande. Mesmo assim, o *software* é extremamente escalável, e pode ser usado para controlar praticamente qualquer parâmetro mensurável em *hardware*, tais como temperatura e umidade (quando suportado). O desenvolvimento da ferramenta é constante e ela possui uma rede grande de usuários que compartilham suas experiências em diversos fóruns espalhados pela *Internet*.

Na Figura 6, observa-se o monitoramento de alguns serviços pelo Cacti.

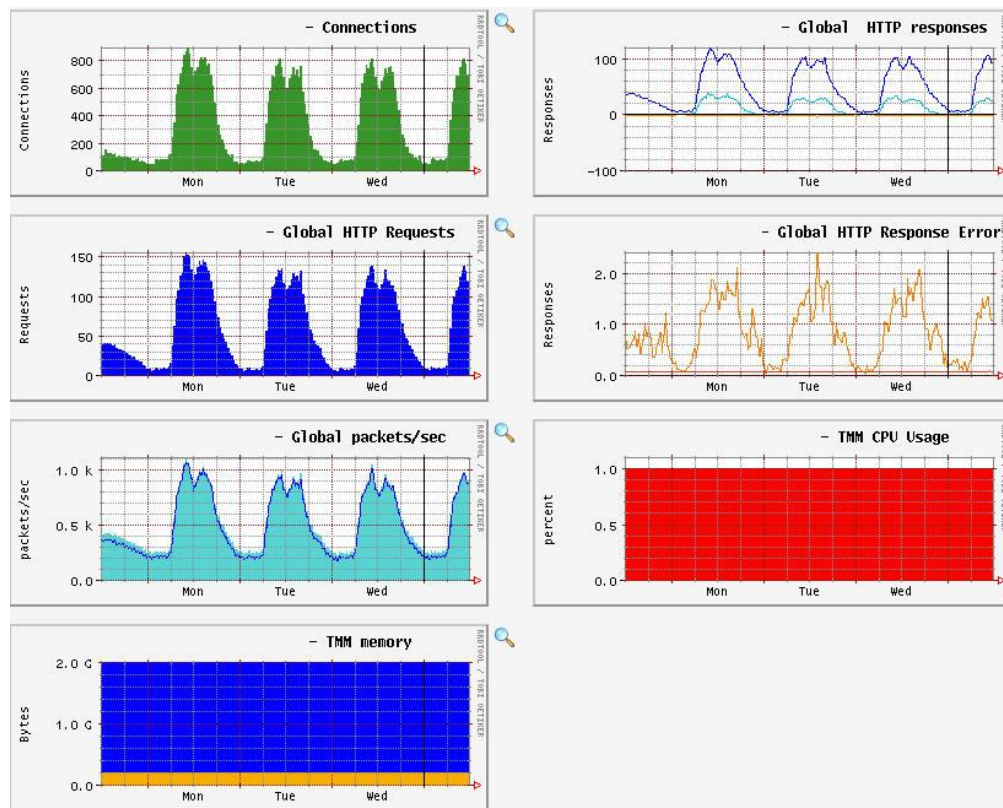


Figura 6 - Monitoramento de alguns serviços pelo Cacti
Fonte: Rahm (2007).

4.5 NAGIOS

De acordo com Lopes (2010), o Nagios é um aplicativo de monitoramento de sistemas e de redes, podendo ser estendido amplamente a um gerenciador de redes graças aos diversos *plug-ins* disponíveis em sua comunidade. Ele verifica clientes e serviços especificados, gerando alertas quando algo está fora dos padrões pré-

definidos. Originalmente desenvolvido para rodar em Linux, há pacotes personalizados para distribuições comuns como Fedora, Ubuntu, SUSE e Debian.

Algumas das várias ferramentas do Nagios TM incluem:

- Monitoramento de rede e serviços;
- Monitoramento dos recursos de clientes (carga de processador, uso de disco, etc.);
- Organização simples de *plugins* que permite aos usuários facilmente desenvolverem seus próprios serviços de checagem;
- Checagem paralela de serviços;
- Habilidade para definir hierarquia de redes de clientes usando clientes pais (*parent hosts*), permitindo a detecção e distinção entre clientes que estão desativados e aqueles que estão inalcançáveis;
- Notificação de contatos quando problemas em serviços e clientes ocorrerem ou forem resolvidos (*via e-mail, pager*, ou métodos definidos pelo usuário);
- Habilidade para definir tratadores de eventos (*event handlers*) que serão executados durante eventos de serviços ou clientes na tentativa de resolução de problemas;
- Rotação automática de arquivos de logs;
- Suporte para implementação de clientes de monitoramento redundantes;
- Interface *web* opcional para visualização do status atual da rede, histórico de notificações e problemas, arquivos de log, etc;

A única exigência para rodar o Nagios é ter um computador rodando Linux (ou variantes do UNIX) e um compilador C, além de ter, evidentemente, a pilha TCP/IP instalada, já que a maioria das checagens de serviços será feita através da rede. Não é obrigatório usar os CGIs incluídos com o Nagios por padrão, mas se optar por usá-los, os seguintes programas serão necessários:

1. Um servidor *web* (preferencialmente Apache);
2. Gd library de Thomas Boutell versão 1.6.3 ou superior (exigido pelos CGIs *statusmap* e *trends*).

O Nagios é distribuído sob os termos da GNU *General Public License* Versão 2, publicado pela *Free Software Foundation*, popularmente conhecido apenas por GPL, garante permissão de copiar, distribuir e modificar o produto sob certas condições. Condições estas especificadas no arquivo '*LICENSE*' que vem na

distribuição do *software* ou acessível online no *site* www.nagios.org. O Nagios é fornecido sem qualquer garantia de qualquer tipo, incluindo a garantia de desenho, mercantibilidade e adequação para um propósito particular. (COSTA, 2008).

Uma vez instalado, existem muitos arquivos de configurações que serão necessários criar ou editar antes de iniciar o monitoramento da rede.

Na Figura 7, observa-se o monitoramento de alguns serviços pelo Nagios.

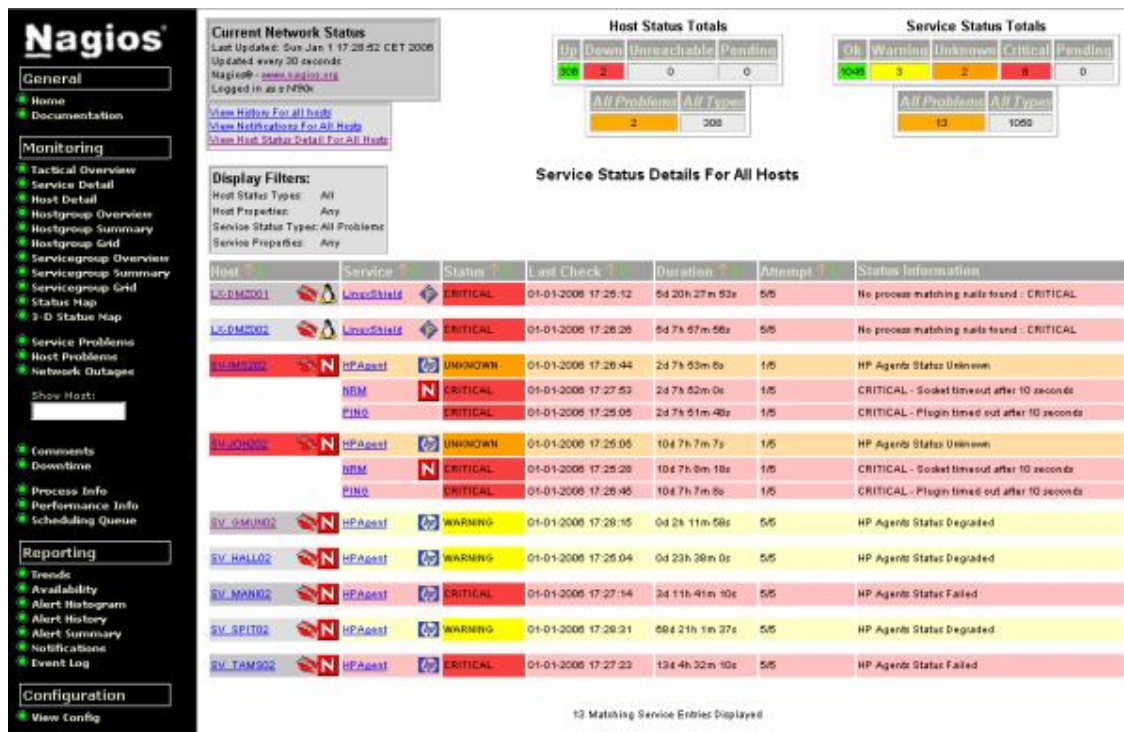


Figura 7 - Monitoramento de alguns serviços pelo Nagios
 Fonte: Lopes (2010).

5 METODOLOGIA

5.1 TIPO DE PESQUISA

O trabalho envolveu uma pesquisa exploratória, pois de acordo com Gil (2008), o objetivo desta é familiarizar-se com um assunto ainda pouco conhecido, pouco explorado. Ao final de uma pesquisa exploratória, se conhecerá mais sobre aquele assunto, estando apto a construir hipóteses. Como qualquer exploração, a pesquisa exploratória depende da intuição do explorador (neste caso, da intuição do pesquisador).

Por ser um tipo de pesquisa muito específica, quase sempre ela assume a forma de um estudo de caso. Como qualquer pesquisa, ela depende também de uma pesquisa bibliográfica, pois mesmo que existam poucas referências sobre o assunto pesquisado, nenhuma pesquisa hoje começa totalmente do zero. Haverá sempre alguma obra, ou entrevista com pessoas que tiveram experiências práticas com problemas semelhantes ou análise de exemplos análogos que podem estimular a compreensão.

5.2 MATERIAIS

O desenvolvimento do trabalho se deu em um microcomputador pessoal (*notebook*) com o sistema operacional Windows XP, equipado com um processador Intel *Dual Core* de 1.8 GHz, 2 Gb de memória RAM, 160 Gb de Disco Rígido, sendo ele emulado em uma máquina virtual com o sistema operacional Linux Ubuntu 10 Server, pois os programas que foram utilizados são compatíveis com tal sistema.

Os instrumentos que foram utilizados, ou seja, as ferramentas foram os *softwares* livres de monitoramento de redes chamados Cacti e Nagios, já apresentados e definidos nos capítulos anteriores. Ambos foram adquiridos através de comandos digitados dentro do terminal do Linux que automaticamente fizeram o *download* dos programas para a máquina virtual através da *Internet*.

A escolha das ferramentas foi feita analisando a funcionalidade e facilidade de uso, e também de serem licenciadas pela GPL (GNU General Public License), ou seja, são *softwares* livres, o que garante um constante desenvolvimento por parte

dos criadores, além de não perder em nada para as soluções comerciais existente, que são extremamente caras.

O objetivo do trabalho foi o de mostrar as funcionalidades e características de cada ferramenta, podendo assim um administrador de rede ter uma visão mais abrangente e ao mesmo tempo uma base de escolha das ferramentas de acordo com suas necessidades.

5.3 PROCEDIMENTOS

De início, foram feitos alguns estudos sobre as características, bem como um aprofundamento no conhecimento dos *softwares* livres Cacti e Nagios através de livros e material disponível na Internet.

Após o estudo das ferramentas, deu-se o início da parte prática do trabalho onde o primeiro passo envolveu a criação da máquina virtual através do programa Sun Virtualbox no computador pessoal. O sistema operacional usado na máquina virtual foi o Linux Ubuntu versão 10.

Em seguida, foi feita a instalação do protocolo SNMP no ambiente Linux pelo fato deste ser o protocolo mais utilizado para o gerenciamento de redes IP e internet.

Após realizado todos os processos descritos acima, foram feitas as instalações e configurações dos *softwares* Cacti e Nagios pelo terminal do Linux que realizou o *download* automático através de comandos digitados no mesmo, instalação essa que utilizou o Arquivo Fonte (*source*) de cada uma.

Foram instalados também algumas dependências dos *softwares* para seu correto funcionamento como o banco de dados MySQL para servir de base na criação e armazenamento dos gráficos e informações geradas pelos *softwares*, garantindo um longo armazenamento dos dados; além do servidor Apache 2 e o Php 5 devido aos *softwares* utilizarem códigos de programação da linguagem Php.

Terminado o processo de instalação e configuração das ferramentas, se iniciou a fase de análise e obtenção dos dados que cada ferramenta gerou de acordo com suas características. Os dados obtidos pelas ferramentas foram guardados e analisados de acordo com o propósito do trabalho.

Por fim, todos os resultados e conclusões foram escritos e documentados para divulgação, atingindo assim a meta principal do projeto.

6 RESULTADOS OBTIDOS

A análise apresentada a seguir advém dos resultados obtidos ao longo do processo de aplicação dos métodos citados anteriormente envolvendo uma série de visualizações e explicações das funcionalidades testadas do Cacti e Nagios.

Inicialmente, foi criada a máquina virtual com o sistema operacional Linux Ubuntu versão 10. O programa utilizado para emular a máquina virtual foi o Sun Virtualbox.

Após realizada a criação do sistema operacional virtual, iniciou-se a instalação e configuração dos softwares através de comandos digitados dentro do terminal do Linux.

Foi trabalhada em primeiro lugar a instalação e configuração do Cacti, onde logo abaixo são mostrados alguns comandos para iniciação do software.

Antes de instalar o Cacti é necessário realizar a instalação de suas dependências.

apt-get install build-essential

Este pacote contém uma lista informativa de pacotes que são considerados essenciais ("build-essential") para a construção de pacotes Debian. Este pacote também depende dos pacotes dessa lista para facilitar a instalação dos pacotes "build-essential".

apt-get install rcconf

Este é um front-end para o comando update-rc. Permite controlar serviços que serão iniciados automaticamente no sistema operacional.

apt-get install libncurses5-dev

Ncurses é uma biblioteca que provê uma API para o desenvolvimento de interfaces em modo texto.

apt-get install libgd2-xpm

Biblioteca de código-fonte aberto para a criação de imagens dinâmicas.

apt-get install libxpm-dev

Libxpm-dev consiste em um formato de imagem do ASCII e de uma biblioteca em C.

apt-get install libpng12-dev

Libpng12-dev é uma biblioteca de referência de imagens PNG.

apt-get install libgdbm-dev

Libgdbm-dev é uma sequência de rotinas de banco de dados que utilizam hash extensivo.

apt-get install snmp**# apt-get install snmpd**

Instalação da dependência SNMP.

apt-get install apache2 apache2-utils

Instalação do Apache 2, ele será utilizado como servidor para o Cacti, visto que ele roda na Web. Ele é necessário para poder executar o Cacti e seus plugins, devido eles serem feitos em php.

apt-get install php5

Instalação do Php 5, Ele é necessário para poder executar o Cacti e seus plugins, devido eles serem feitos em php.

apt-get install libapache2-mod-php5

Módulo de integração do Apache e PHP.

apt-get install mysql-server

Instalação do bancos de dados MySQL.

apt-get install cacti**# apt-get install cacti-spine**

Instalação do Cacti

Após feito o procedimento de instalação, é digitado no navegador o endereço `http://localhost/cacti/` para realizar o login mediante validação de usuário e senha conforme Figura 8.

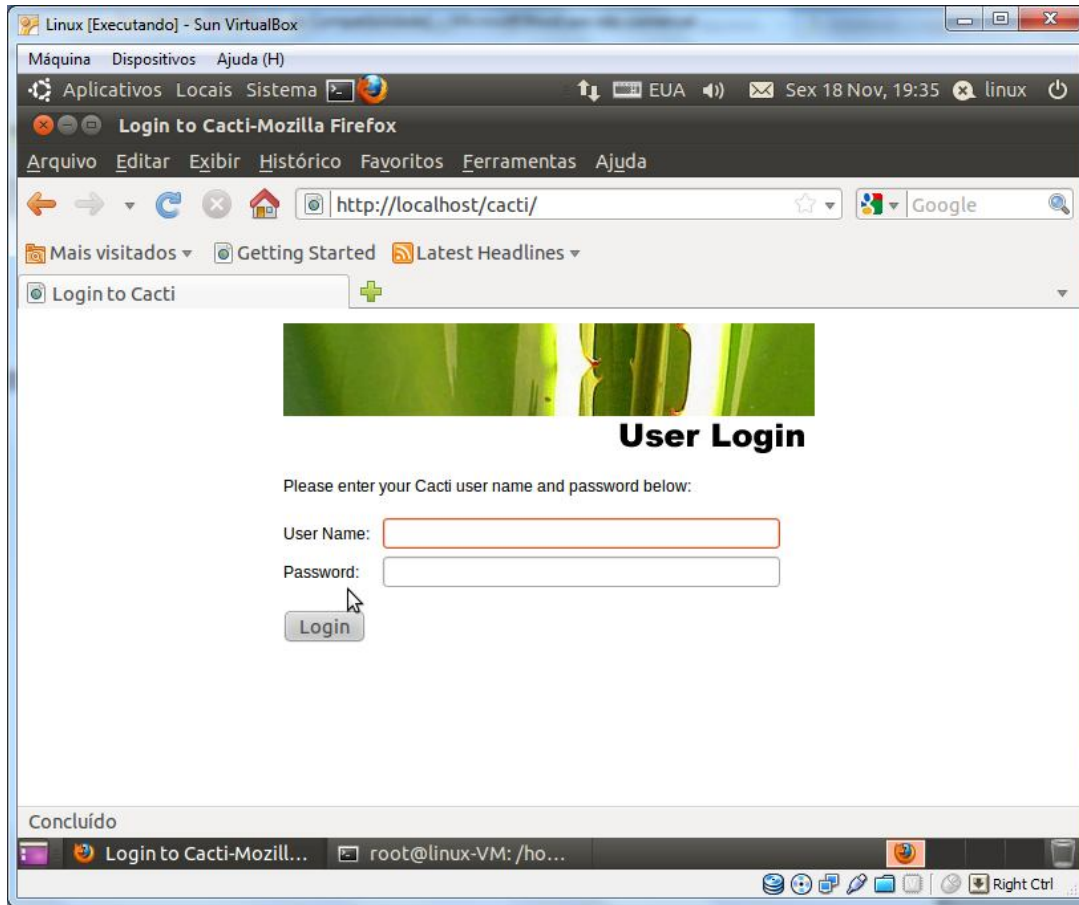


Figura 8 - Tela de Login do Cacti

Na figura 9, é possível visualizar a tela inicial do Cacti onde pode-se acessar suas abas principais de recursos e adicionar dispositivos para monitoramento. Juntamente com esses dispositivos podem ser criados gráficos para visualização de estatísticas sobre o gerenciamento.

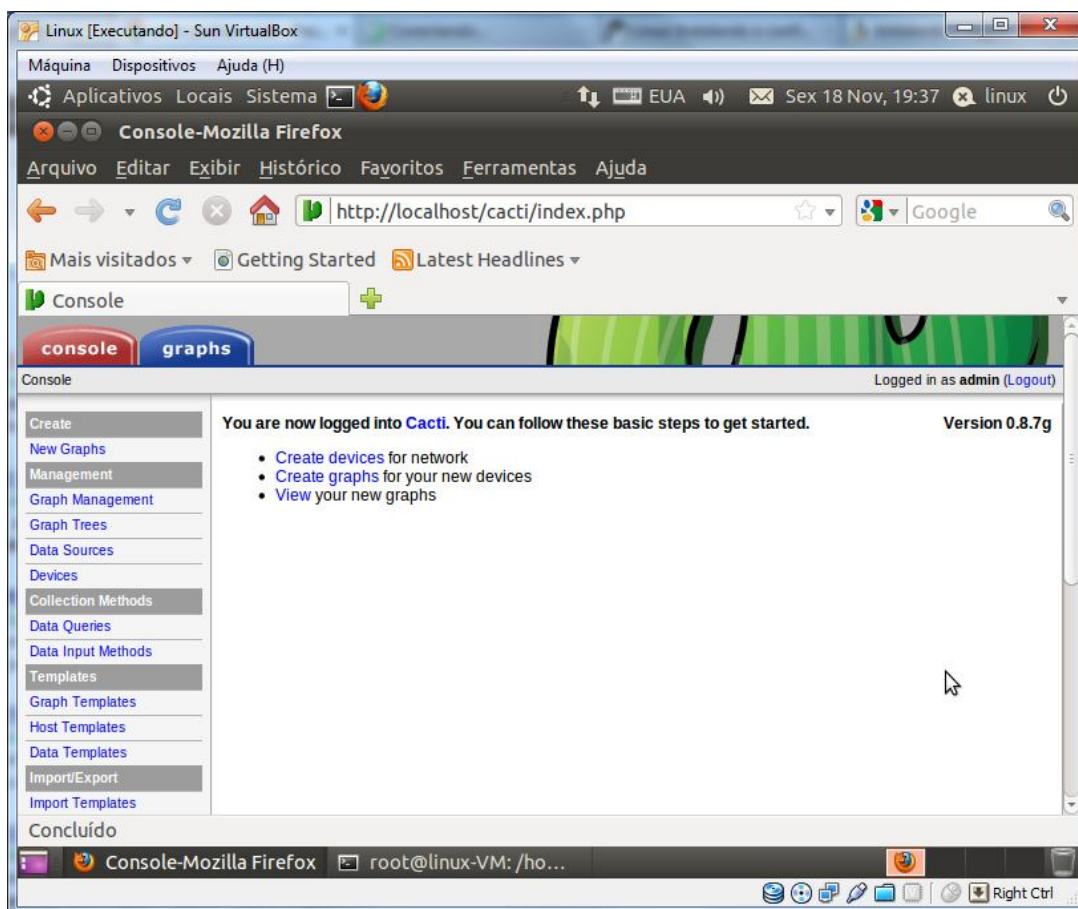


Figura 9 - Tela inicial do Cacti.

Já na Figura 10, uma das funcionalidades do Cacti, o uso de memória, é mostrado graficamente. Ele possui um sistema de monitoramento com marcações legendadas do dia e das horas monitoradas, além de detalhar a quantidade em kilobytes usada pela memória.

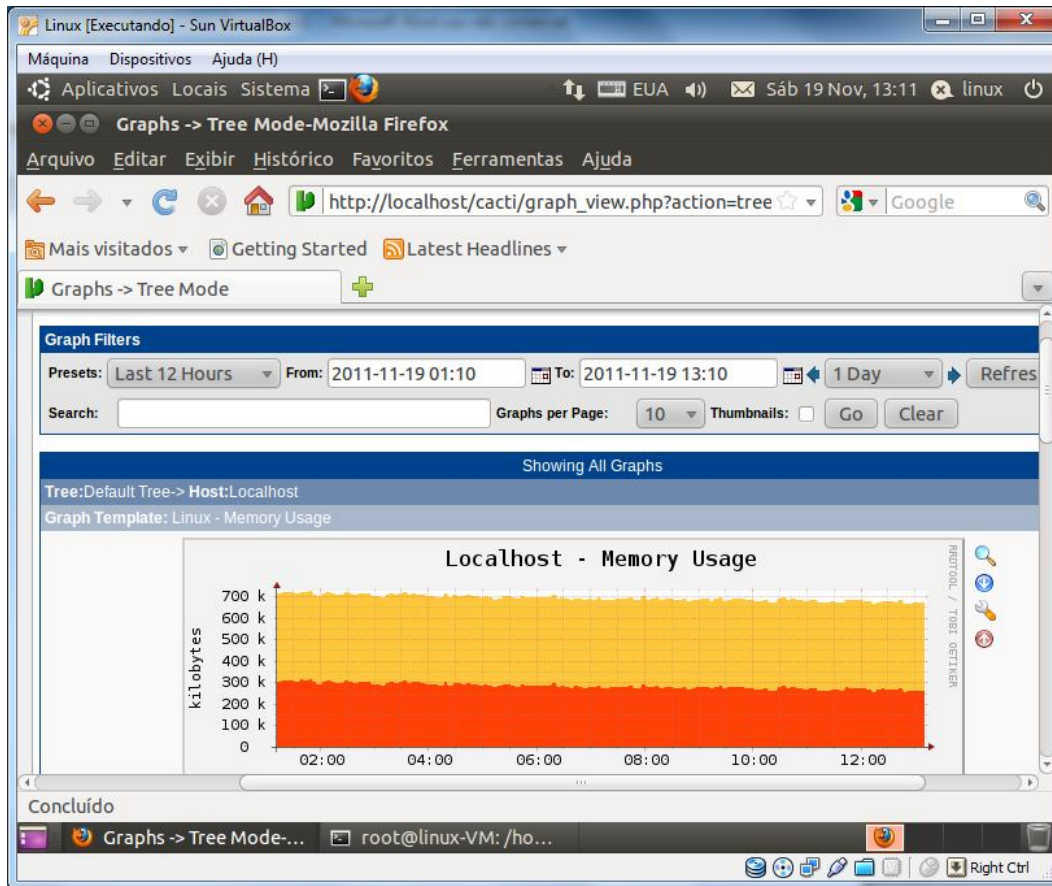


Figura 10 - Gráfico do uso de memória gerado pelo Cacti.

Na Figura 11, outra das funcionalidades do Cacti, a média de carregamento, é mostrada graficamente. Ele possui um sistema de monitoramento com marcações legendadas do dia e das horas monitoradas, além de detalhar a média de tempo em que os serviços são carregados no sistema.

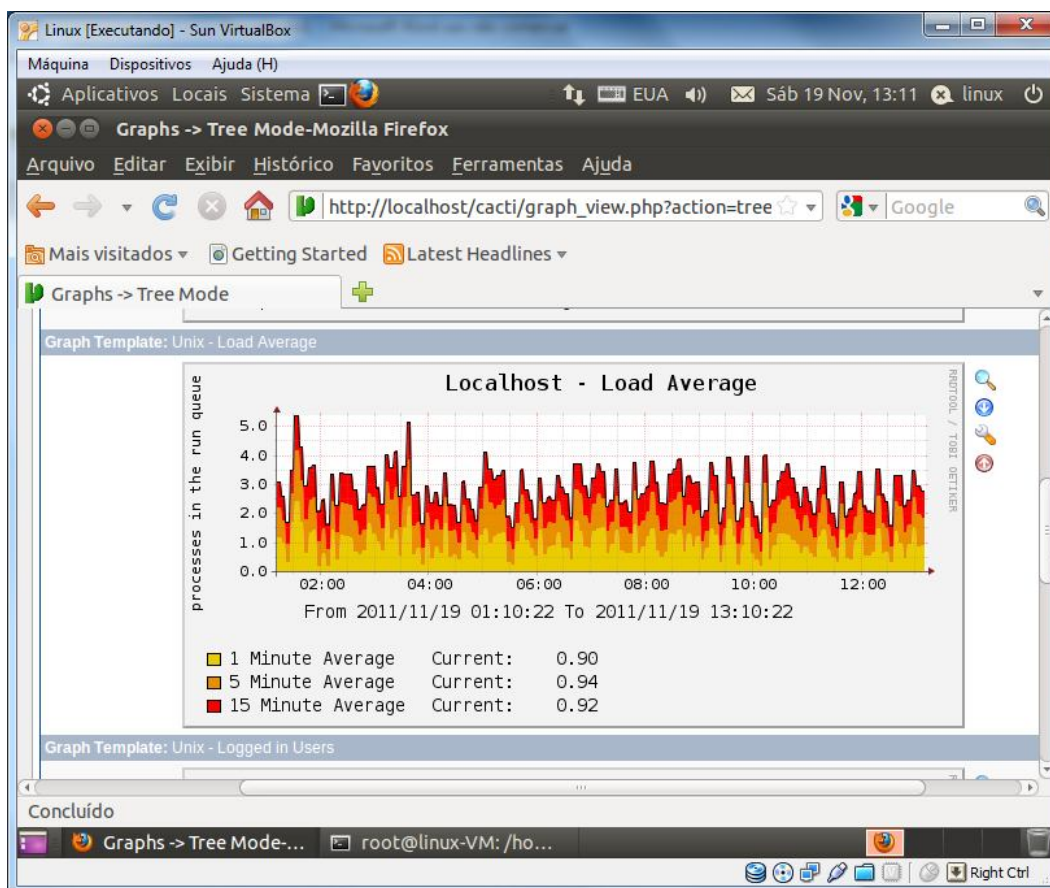


Figura 11 - Gráfico da média de carregamento gerado pelo Cacti.

Na Figura 12, outra das funcionalidades do Cacti, usuários logados, são mostrados graficamente. Ele possui um sistema de monitoramento com marcações legendadas do dia e das horas monitoradas, além de detalhar a quantidade de usuários que estão logados no sistema em determinado período de tempo.

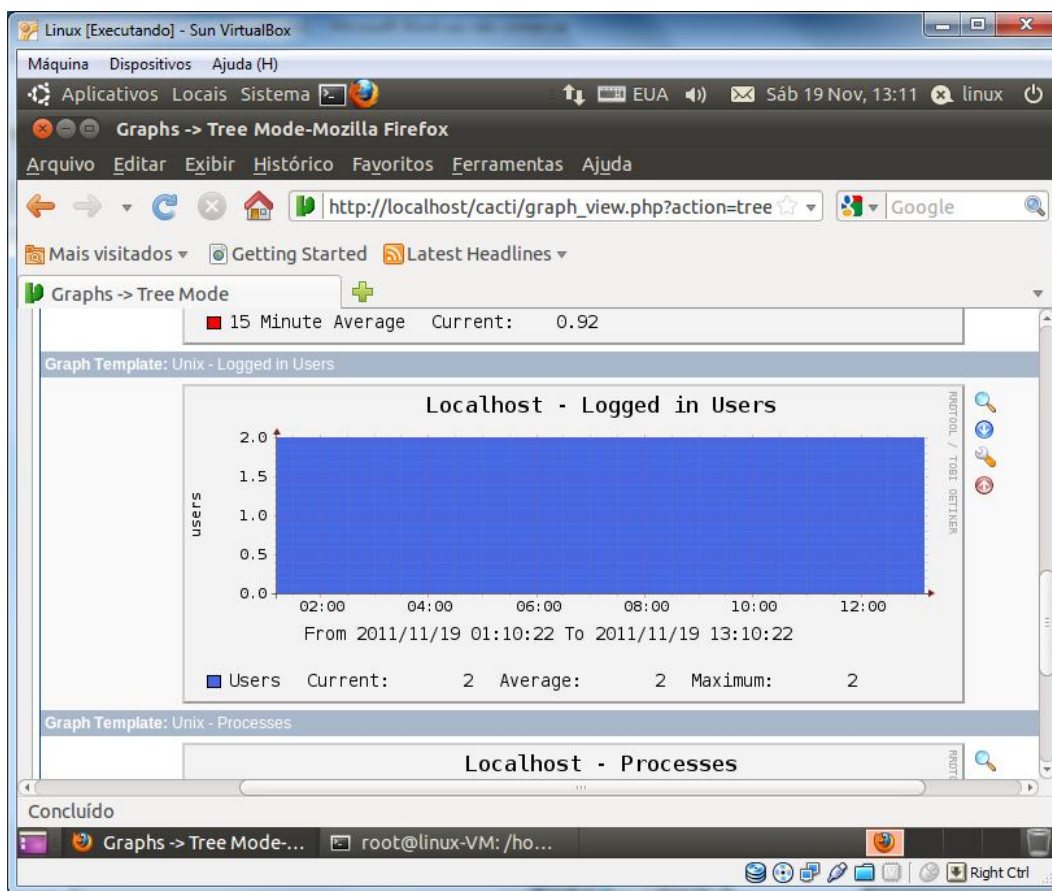


Figura 12 - Gráfico de usuários logados gerado pelo Cacti.

Na Figura 13, mais uma das funcionalidades do Cacti, processos, são mostrados graficamente. Nele, são mostrados a quantidade de processos sendo executados na rede por um sistema de monitoramento com marcações legendadas do dia e das horas monitoradas.

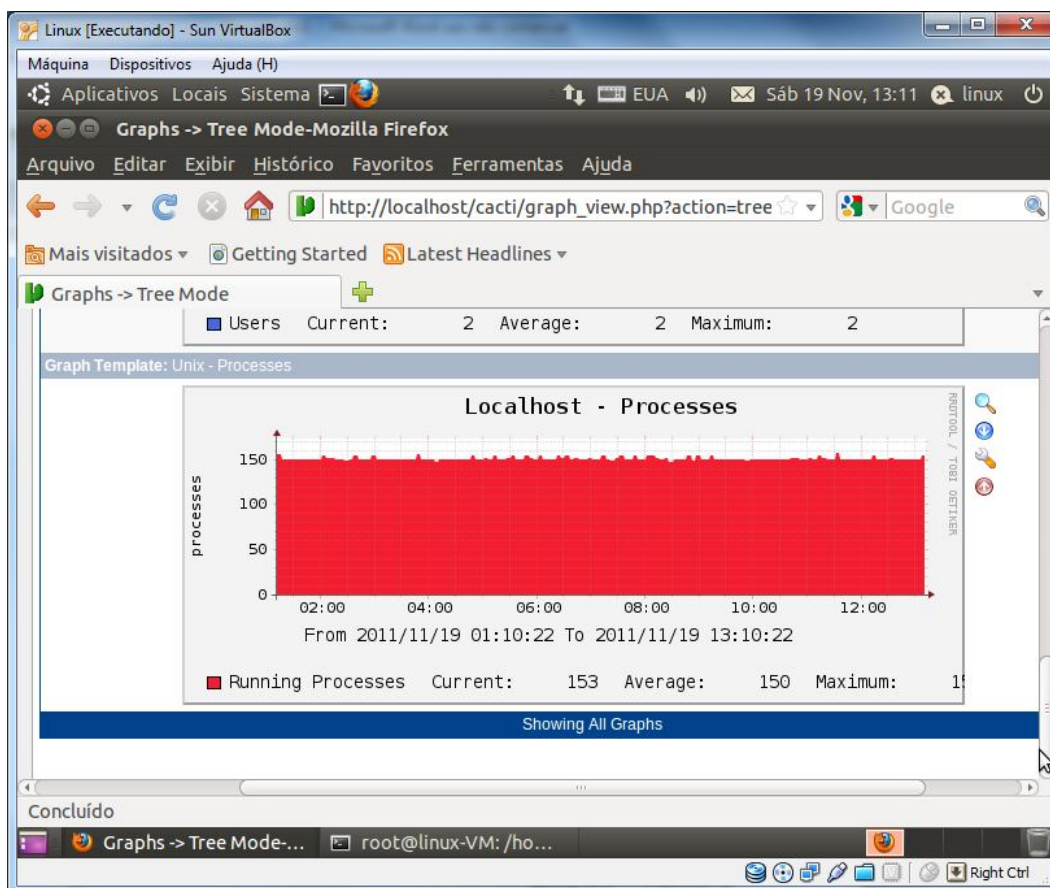


Figura 13 - Gráfico de processos gerado pelo Cacti.

Todos esses gráficos e dispositivos mostrados anteriormente podem ser configurados de acordo com os recursos disponíveis para o usuário conforme mostra a Figura 14.

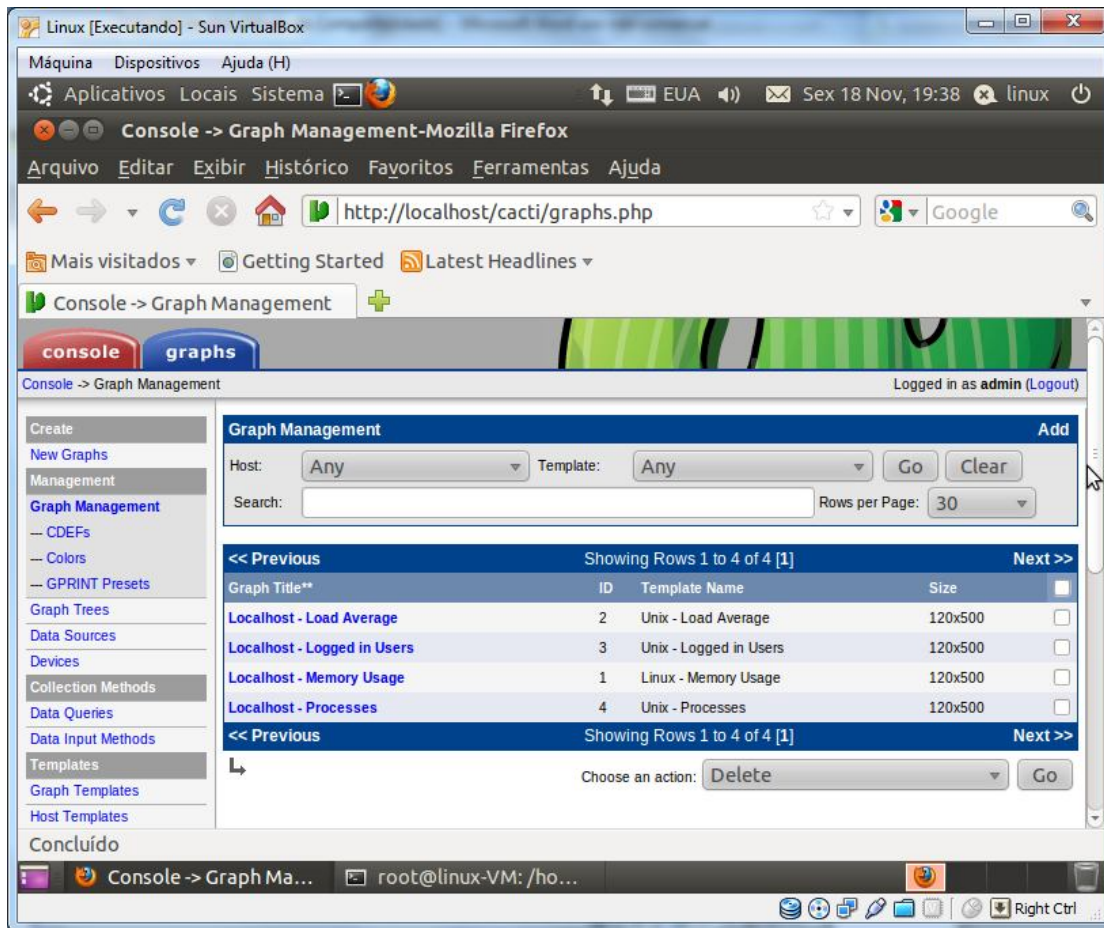


Figura 14 - Tela para gerenciar o computador que está sendo monitorado.

Em seguida, na Figura 15, é apresentada a tela onde são adicionados e monitorados os dispositivos desejados pelo administrador.

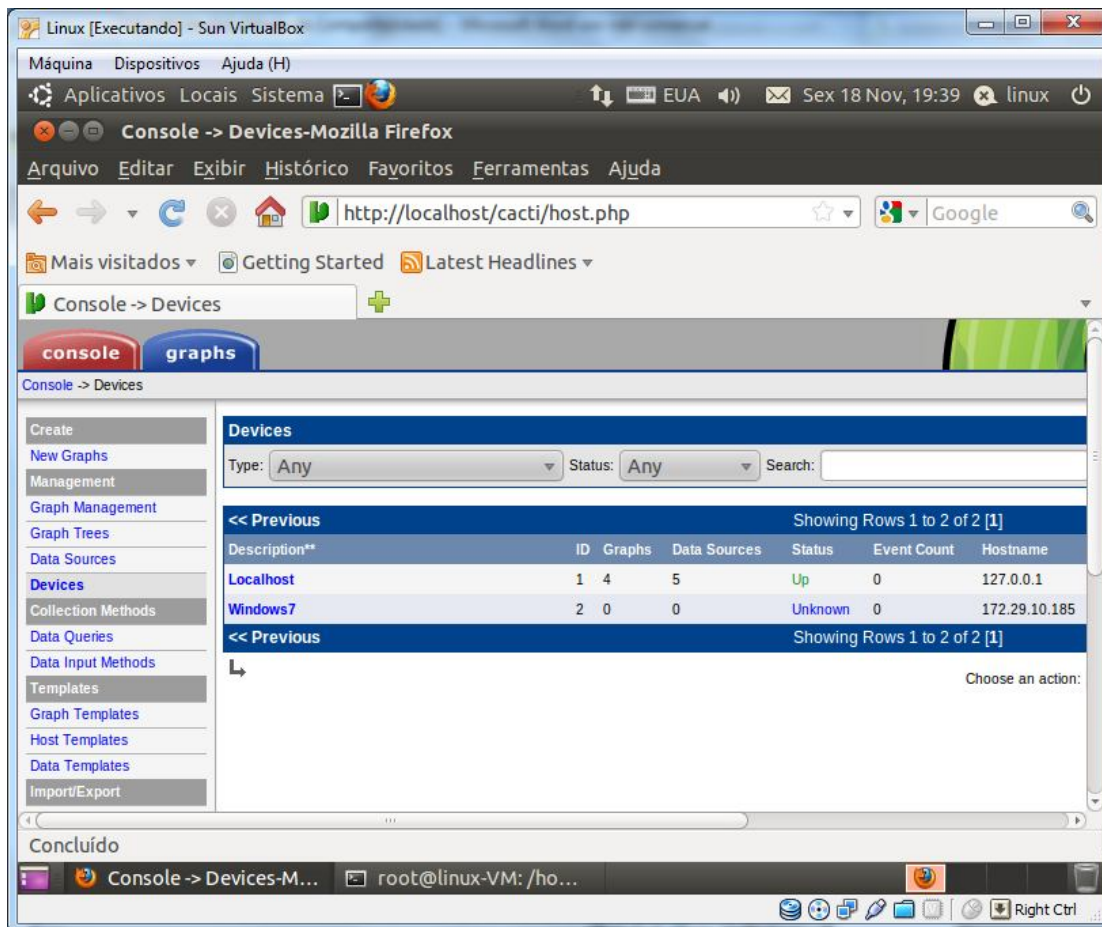
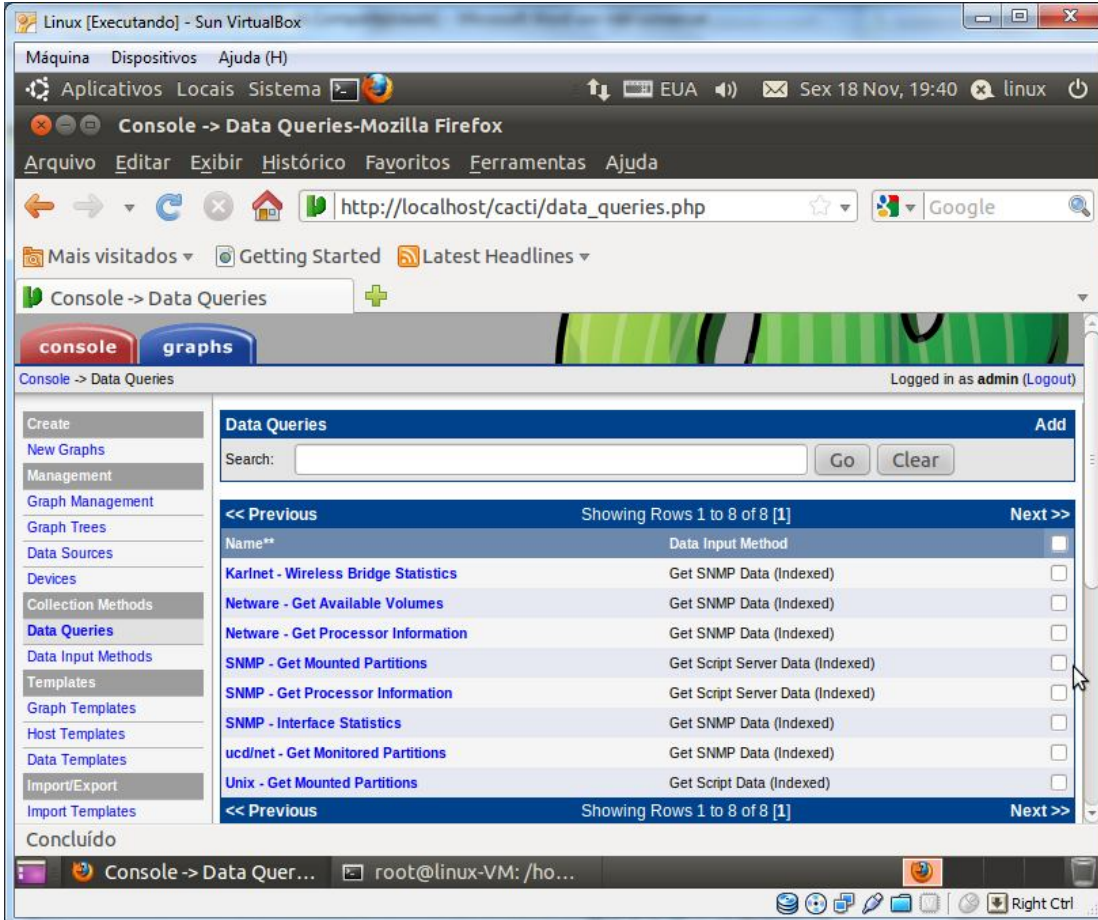


Figura 15 - Dispositivos sendo monitorados pelo Cacti.

A Figura 16, apresenta o serviço para a consulta dos dados e estatísticas processadas pelo Cacti, entre eles, as informações obtidas e geradas através do protocolo SNMP.



The screenshot shows a web browser window displaying the Cacti Data Queries page. The browser's address bar shows the URL `http://localhost/cacti/data_queries.php`. The page title is "Console -> Data Queries" and it indicates the user is logged in as "admin".

The page content includes a sidebar with navigation options such as "Create", "New Graphs", "Management", "Graph Management", "Graph Trees", "Data Sources", "Devices", "Collection Methods", "Data Queries", "Data Input Methods", "Templates", "Graph Templates", "Host Templates", "Data Templates", "Import/Export", and "Import Templates".

The main content area is titled "Data Queries" and features a search bar with "Go" and "Clear" buttons. Below the search bar is a table listing various data queries. The table has two columns: "Name**" and "Data Input Method". There are 8 rows of data, each with a checkbox in the right margin.

Name**	Data Input Method	
Karlnet - Wireless Bridge Statistics	Get SNMP Data (Indexed)	<input type="checkbox"/>
Netware - Get Available Volumes	Get SNMP Data (Indexed)	<input type="checkbox"/>
Netware - Get Processor Information	Get SNMP Data (Indexed)	<input type="checkbox"/>
SNMP - Get Mounted Partitions	Get Script Server Data (Indexed)	<input type="checkbox"/>
SNMP - Get Processor Information	Get Script Server Data (Indexed)	<input type="checkbox"/>
SNMP - Interface Statistics	Get SNMP Data (Indexed)	<input type="checkbox"/>
ucd/net - Get Monitored Partitions	Get SNMP Data (Indexed)	<input type="checkbox"/>
Unix - Get Mounted Partitions	Get Script Data (Indexed)	<input type="checkbox"/>

At the bottom of the browser window, the Linux terminal shows the prompt `root@linux-VM: /ho...`.

Figura 16 - Consulta de dados pelo Cacti.

Outra opção disponível, conforme Figura 17, é a criação e visualização de modelos gráficos, de *host* e de dados de acordo com as escolhas desejadas pelo administrador.

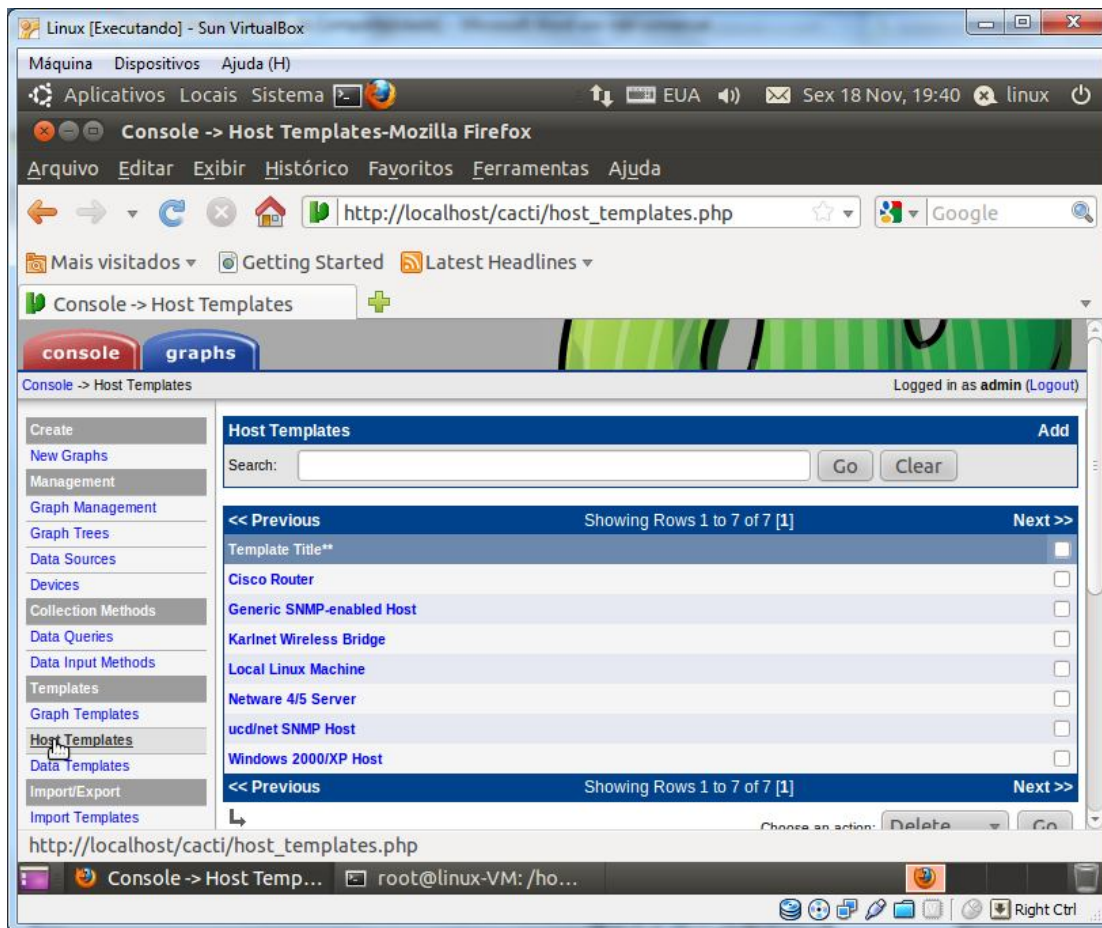


Figura 17 - Visualização de modelos gráficos, host e de dados gerado pelo Cacti.

O Cacti possui também uma grade ampla de configurações para o monitoramento como pode-se observar na Figura 18.

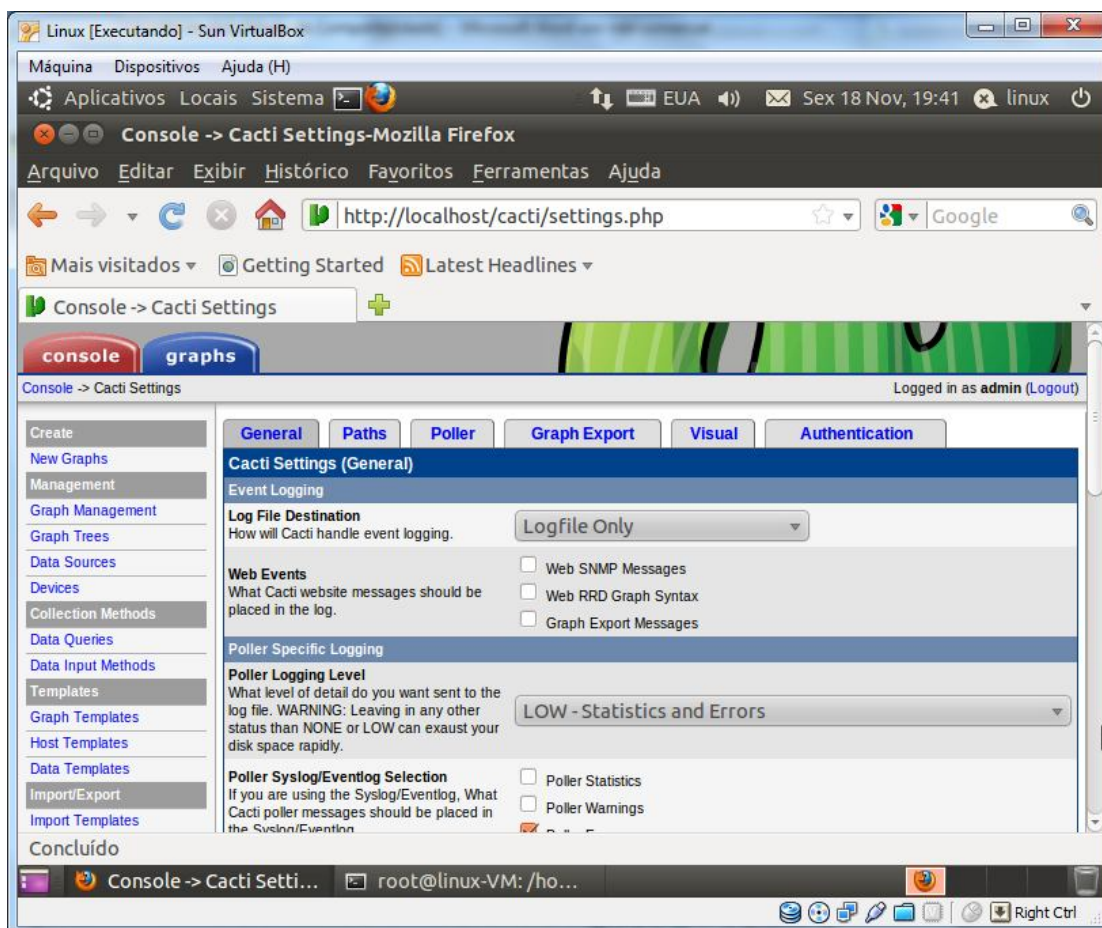


Figura 18 - Configurações gerais disponíveis pelo Cacti.

E por fim, outro dos recursos disponíveis no Cacti, mostrado na Figura 19, é o relatório com informações dos usuários que estão sendo monitorados pelo programa que vão do administrador até os usuários hospedados pela rede.

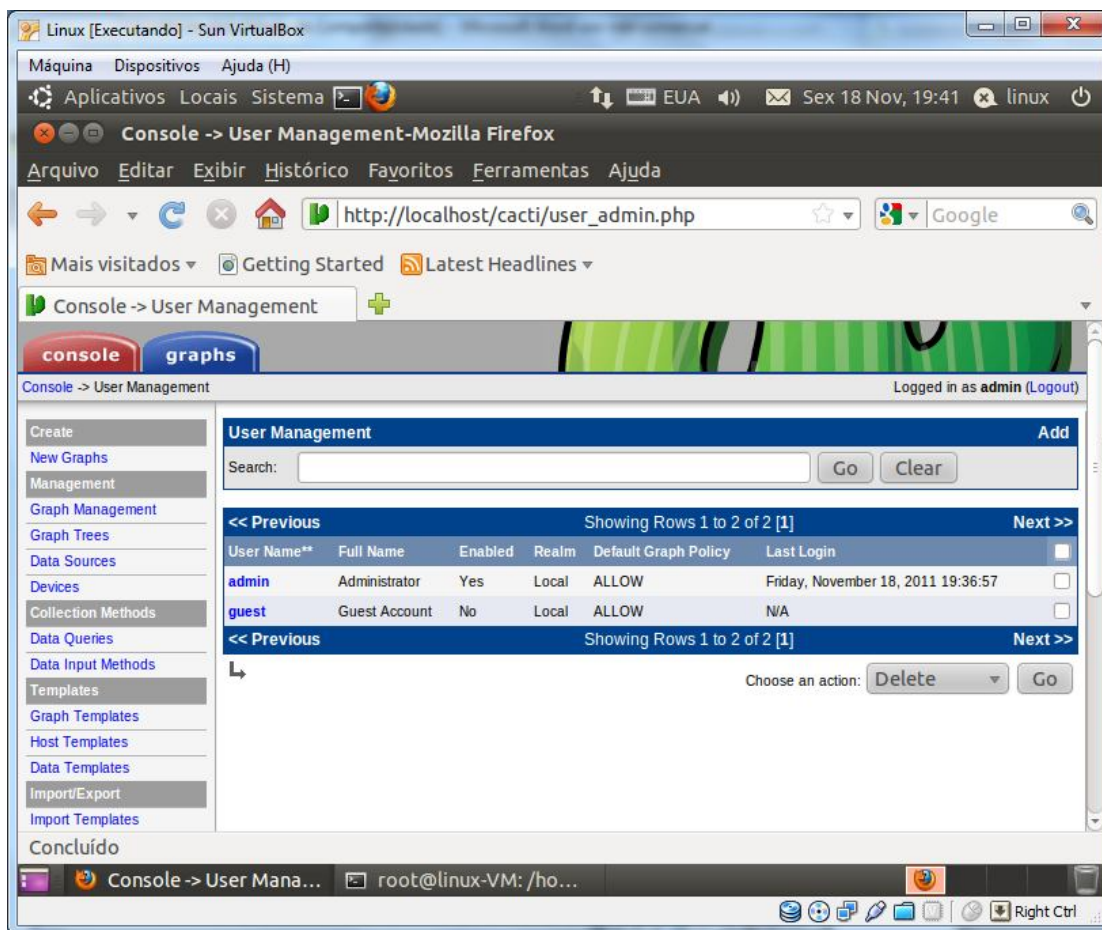


Figura 19 - Usuários monitorados pelo Cacti.

A seguir, foi analisado o software Nagios, mostrando alguns comandos para inicialização do mesmo. Comandos para instalação de dependências do programa:

```
# apt-get -y install openssl (implementa as funções básicas de criptografia)
# apt-get -y install libssl-dev (bibliotecas para criptografia)
# apt-get -y install build-essential (lista de pacotes para compilação)
# apt-get -y install nmap (serviço de sniffer)
# apt-get -y install xinetd (controla os serviços a serem acessados)
# apt-get -y install apache2 (servidor web)
# apt-get -y install libjpeg-dev (bibliotecas para imagem)
```



```
# apt-get -y install libpng12-0 (bibliotecas para imagem)
# apt-get -y install libpng12-dev (bibliotecas para imagem)
# apt-get -y install libgd2-xpm (bibliotecas para gerar gráficos)
# apt-get -y install libgd2-xpm-dev (bibliotecas para gerar gráficos)
# apt-get -y install fontconfig (biblioteca de configuração de fontes genérica)
# apt-get -y install sudo (instalação do super usuário)
```

Após instaladas as dependências, foi baixado e compilado o programa.

Depois de feito o procedimento de instalação, foi digitado no navegador o endereço `http://localhost/nagios/` para realizar o login mediante validação de usuário e senha conforme Figura 20.

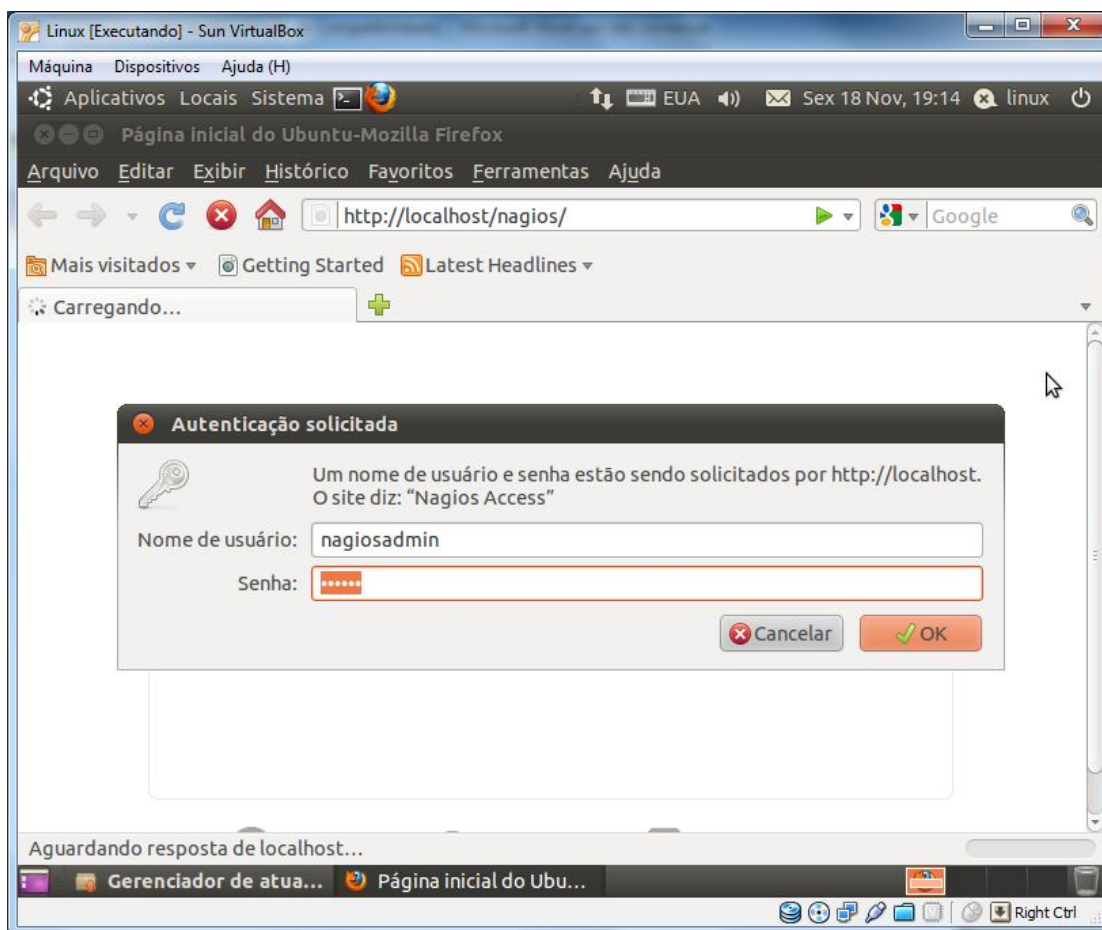


Figura 20 - Tela de login do Nagios.

Na figura 21, é possível visualizar a tela inicial do Nagios onde pode-se acessar suas abas principais de recursos. Nessa aba, vê-se uma série de opções para o monitoramento conforme mostradas nas figuras adiante.

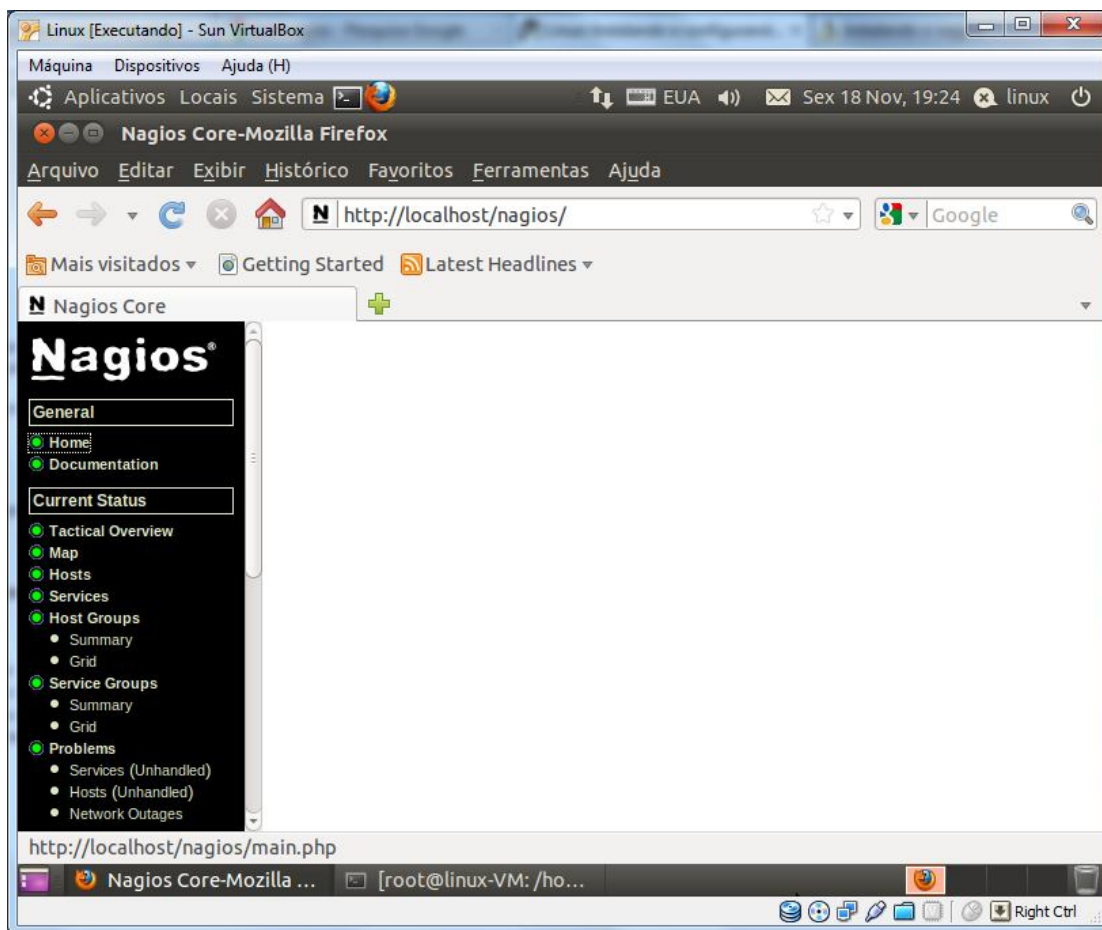


Figura 21 - Tela inicial do Nagios.

Observando as Figuras 22 e 23, é mostrado um resumo geral do monitoramento e sua performance diante da rede analisada com itens bem detalhados, além de fornecer a informação sobre a “saúde” da rede contemplando o *host* e os serviços. Também é possível checar alguns recursos da monitoração como detecção de *flap*, notificações, manipuladores de eventos, verificações ativas.

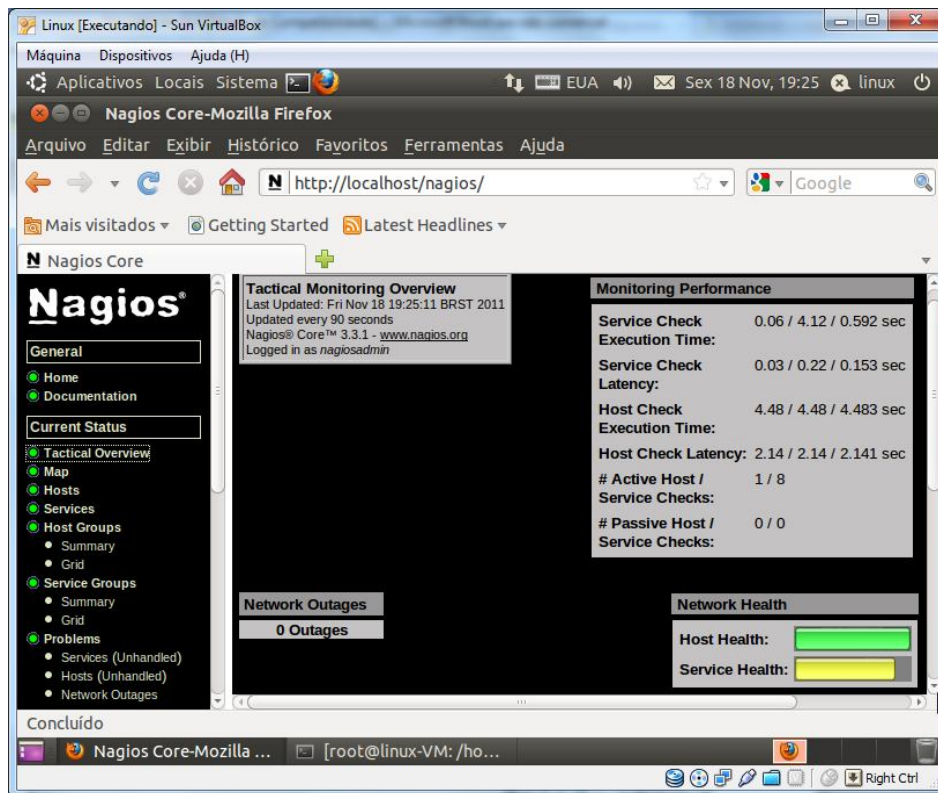


Figura 22 - Resumo geral da performance de monitoramento do Nagios.



Figura 23 - Recursos de monitoramento do Nagios.

Já nas Figuras 24, 25 e 26, o Nagios mostra o detalhamento dos *hosts* e dos serviços que eles utilizam, informando e notificando avisos se houver algum problema com ambos, além de mostrar o período de checagem e o status dos mais variados recursos como, por exemplo, a perda de pacotes.

The screenshot displays the Nagios Core web interface. The browser window title is 'Nagios Core-Mozilla Firefox' and the address bar shows 'http://localhost/nagios/'. The interface includes a sidebar with navigation options like 'Home', 'Documentation', 'Tactical Overview', 'Map', 'Hosts', 'Services', 'Host Groups', 'Service Groups', and 'Problems'. The main content area shows the 'Current Network Status' and two summary tables:

Host Status Totals				Service Status Totals				
Up	Down	Unreachable	Pending	Ok	Warning	Unknown	Critical	Pending
1	0	0	0	7	0	0	1	0

Below the tables are links for 'All Problems' and 'All Types' for both Hosts and Services. The 'Host Status Details For' section is partially visible at the bottom.

Figura 24 - Detalhamento de hosts e serviços do Nagios.

The screenshot shows the Nagios Core web interface in a Mozilla Firefox browser. The page title is "Host Status Details For All Host Groups". The left sidebar contains a navigation menu with sections: General, Current Status, Tactical Overview, Map, Hosts, Services, Host Groups, Service Groups, and Problems. The main content area displays a table with the following data:

Host	Status	Last Check	Duration	Status Information
localhost	UP	11-18-2011 19:23:43	29d 22h 54m 13s	PING OK - Perda de pacotes = 0%, RTA = 1.98 ms

Below the table, it indicates "1 Matching Host Entries Displayed".

Figura 25 - Detalhamento de hosts e serviços do Nagios.

The screenshot shows the Nagios Core web interface displaying detailed service status for localhost. The table below shows the following data:

Host	Service	Status	Last Check	Duration	Attempt	Status Information
localhost	Current Load	OK	11-18-2011 19:24:18	29d 22h 54m 44s	1/4	OK - Carga média: 1.24, 1.40, 1.11
localhost	Current Users	OK	11-18-2011 19:24:56	29d 22h 54m 6s	1/4	USUÁRIOS OK - 1 usuário atualmente logados em
localhost	HTTP	OK	11-18-2011 19:25:33	29d 22h 53m 29s	1/4	HTTP OK: HTTP/1.1 200 OK - 453 bytes em 0,031 segundos no tempo de resposta
localhost	PING	OK	11-18-2011 19:26:11	29d 22h 52m 51s	1/4	PING OK - Perda de pacotes = 0%, RTA = 0.13 ms
localhost	Root Partition	OK	10-19-2011 20:59:24	29d 22h 52m 14s	1/4	DISK OK - free space: / 4248 MB (58% inode=70%)
localhost	SSH	CRITICAL	10-19-2011 20:58:02	29d 22h 51m 36s	4/4	Conexão recusada
localhost	Swap Usage	OK	10-19-2011 20:55:39	29d 22h 50m 59s	1/4	SWAP OK - 100% free (397 MB out of 397 MB)
localhost	Total Processes	OK	10-19-2011 20:56:17	29d 22h 50m 21s	1/4	PROCS OK: 84 processos com

Figura 26 - Detalhamento de hosts e serviços do Nagios.

Nesta parte do programa, observando a Figuras 27, o Nagios permite o acesso à relatórios detalhados em relação ao monitoramento da rede. Esse relatório é gerado através de passos onde o administrador escolhe as opções que lhe melhor convêm. O primeiro passo envolve o tipo de relatório a ser gerado.

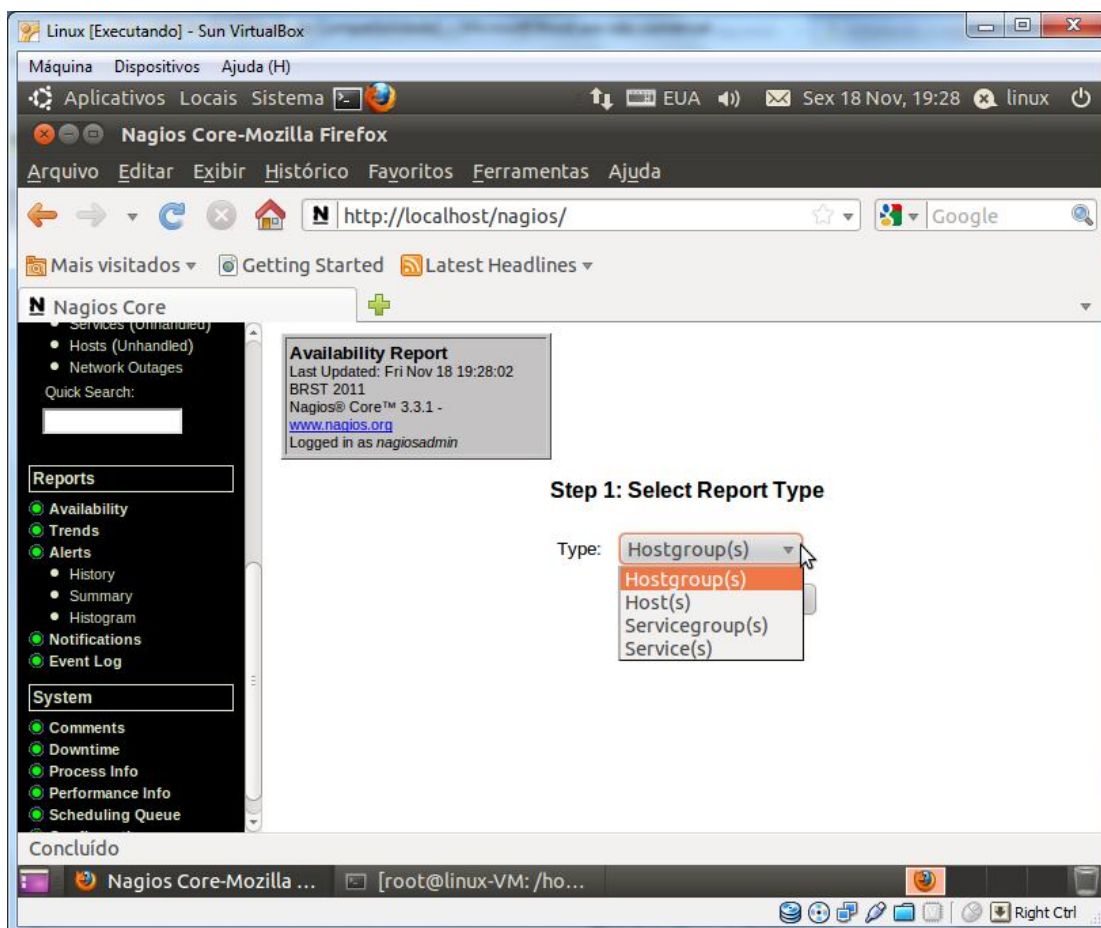


Figura 27 - Relatórios do Nagios – Passo 1.

No segundo passo, conforme Figura 28, são escolhidos os serviços a serem relatados.

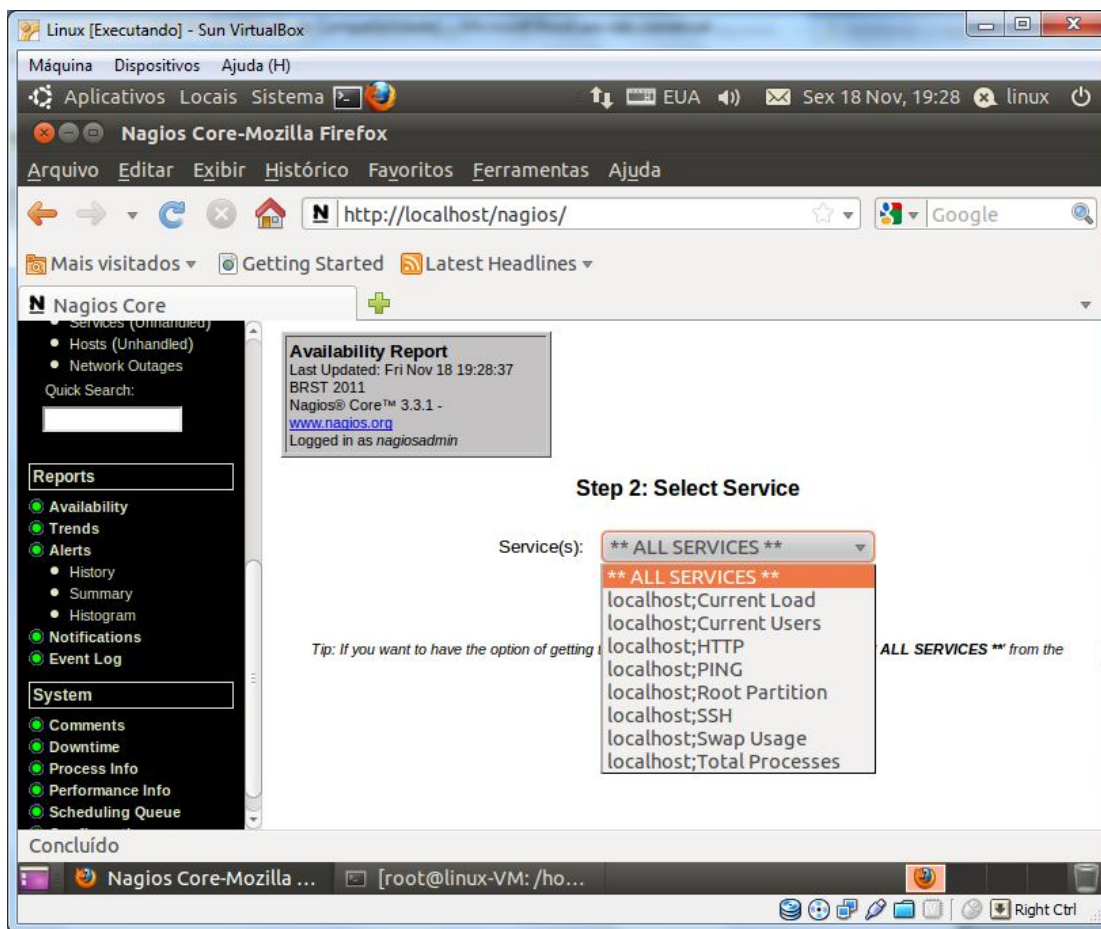


Figura 28 - Relatórios do Nagios – Passo 2.

Já no terceiro passo, conforme Figura 29, é escolhida o período de tempo em que os serviços foram monitorados.

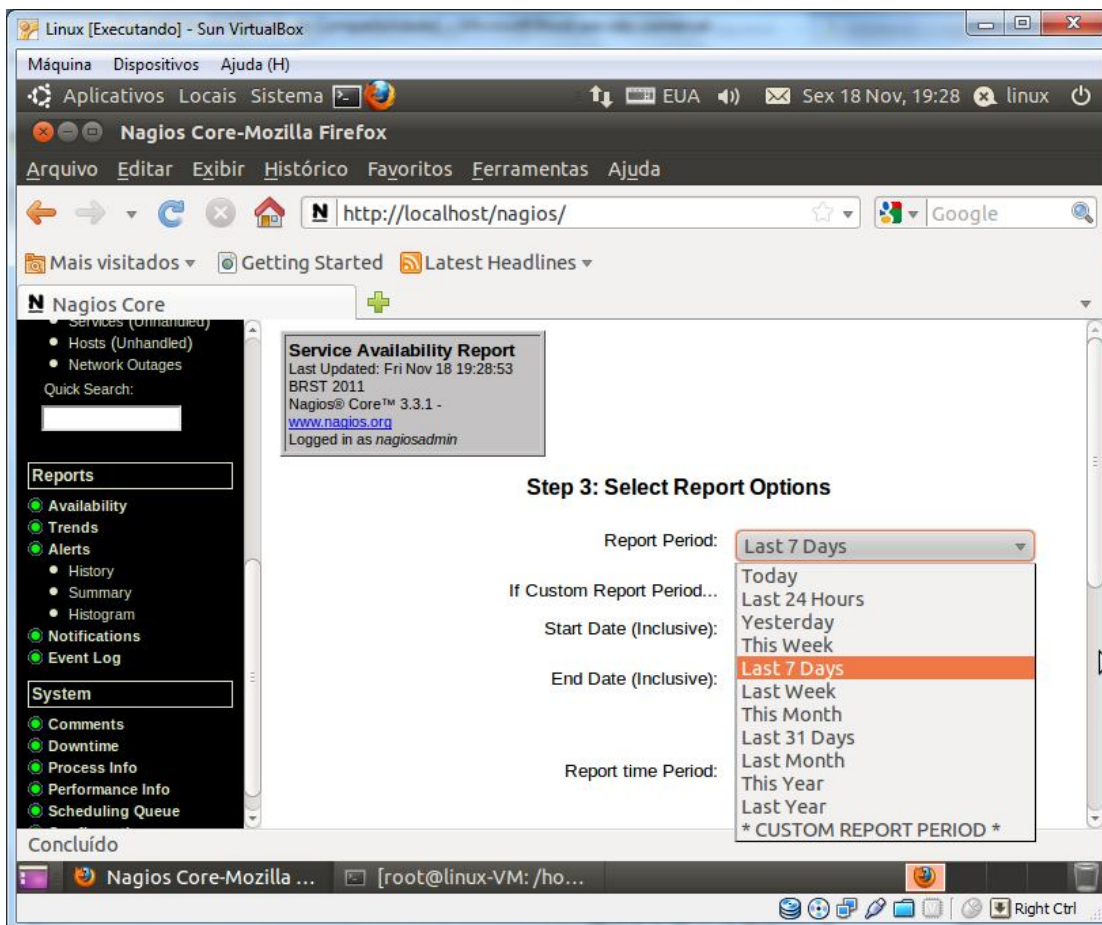


Figura 29 - Relatórios do Nagios – Passo 3.

E por fim, conforme Figura 30, é gerado o relatório com todas as opções escolhidas detalhadas pelo monitoramento.

Service Availability Report
 Last Updated: Fri Nov 18 19:29:08 BRST 2011
 Nagios® Core™ 3.3.1 - www.nagios.org
 Logged in as nagiosadmin

All Services
 First assumed service state: Unspecified
 Report period: Last 7 Days
 Backtracked archives: 4
 Update

[Availability report completed in 0 min 0 sec]

Service State Breakdowns:

Host	Service	% Time OK	% Time Warning	% Time Unknown	% Time Critical	% Time Undetermined
localhost	Current Load	0.000% (0.000%)	0.000% (0.000%)	0.000% (0.000%)	0.000% (0.000%)	100.000%
	Current Users	0.000% (0.000%)	0.000% (0.000%)	0.000% (0.000%)	0.000% (0.000%)	100.000%
	HTTP	0.000% (0.000%)	0.000% (0.000%)	0.000% (0.000%)	0.000% (0.000%)	100.000%

Concluído

Figura 30 - Relatórios do Nagios – Detalhamento.

Uma parte importante do programa, como visto na Figura 31, envolve o gerenciamento de alertas, onde qualquer erro que ocorra com algum serviço ou alguma falha no sistema, como uma máquina na rede desligada, o Nagios emite um alerta (Host Down) avisando sobre o problema.

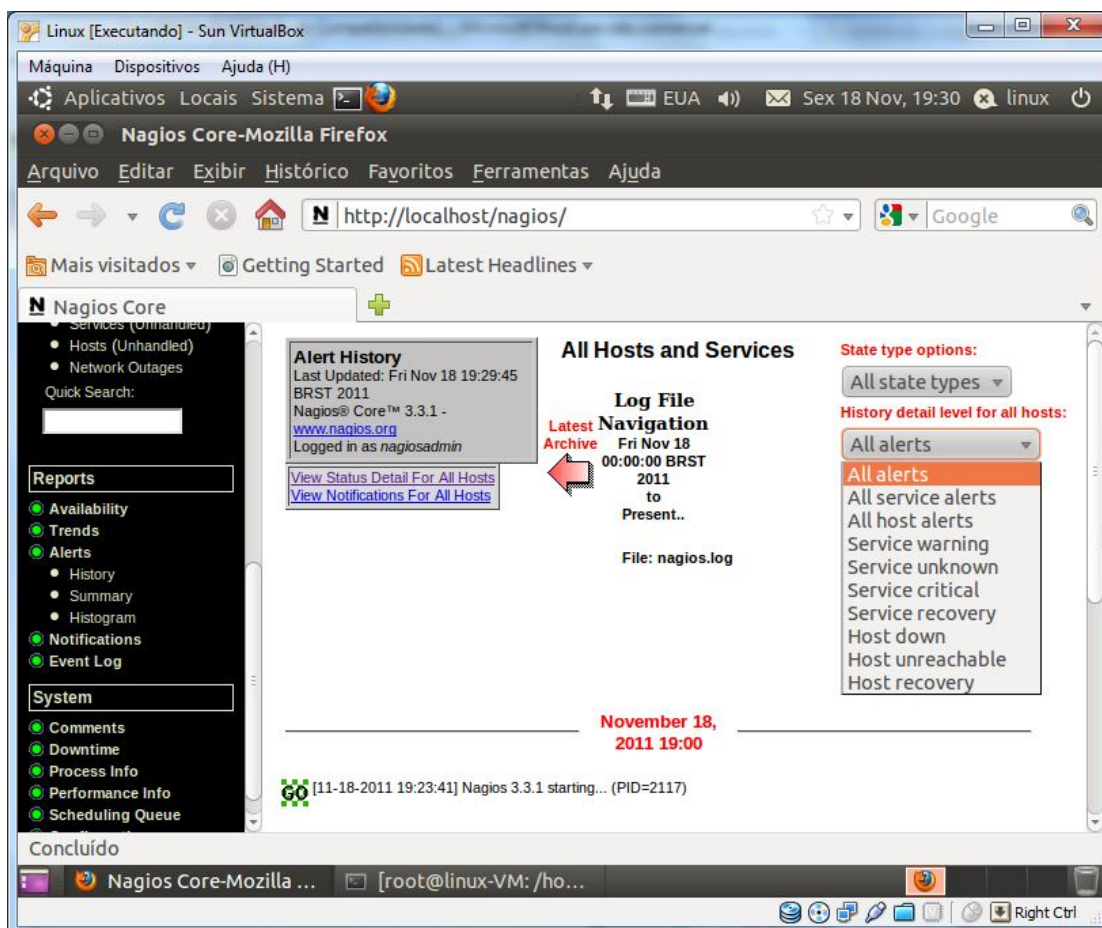


Figura 31 - Gerenciamento de alertas.

A seguir, outro recurso importante do Nagios, conforme mostra Figura 32, os *Event Logs*, ou seja, um sistema de gerenciamento e correlação de todos os eventos trabalhados pelo programa onde qualquer processo executado é registrado numa lista com informações detalhadas.

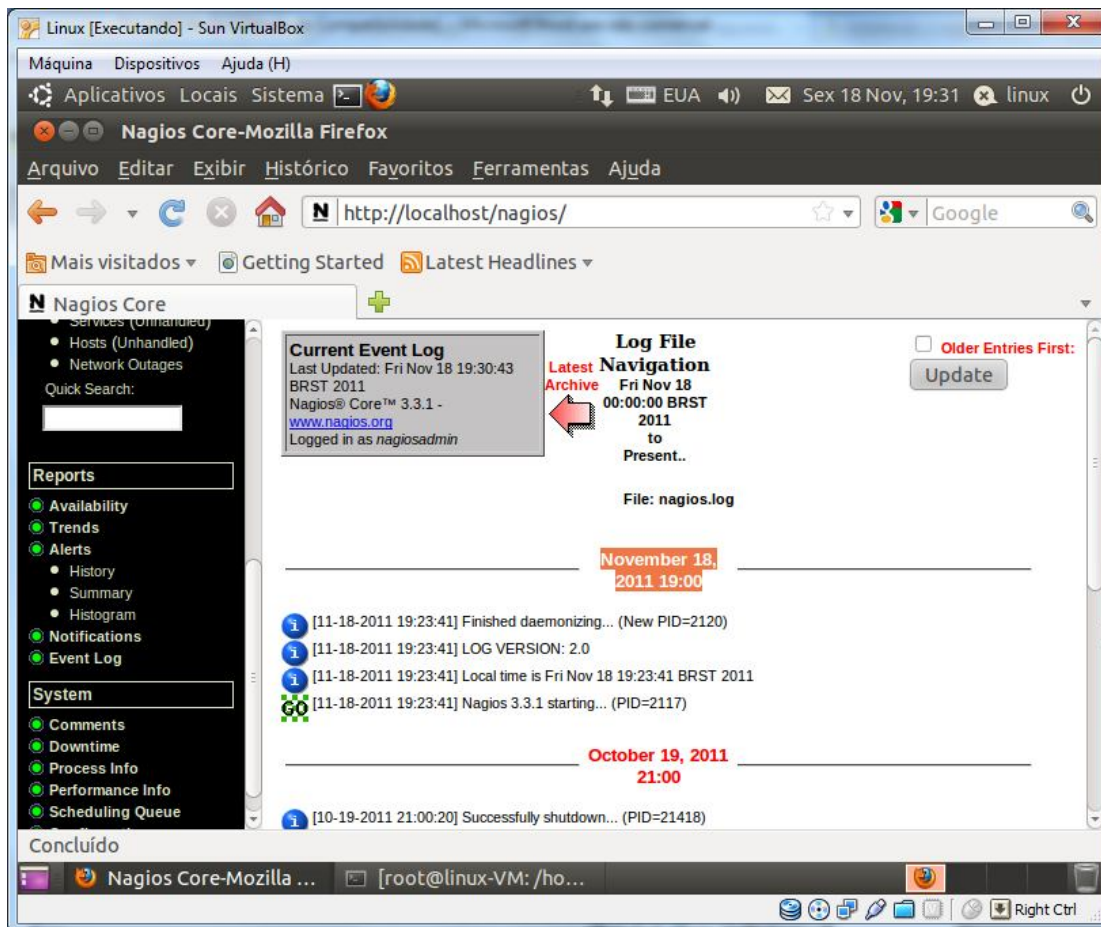


Figura 32 - Registro de eventos do Nagios.

Na Figura 33 são mostrados informações sobre os processos executados com avisos do status de cada opção habilitada juntamente com seus comandos.

The screenshot shows a Linux virtual machine window titled 'Linux [Executando] - Sun VirtualBox'. Inside, a Mozilla Firefox browser window is open to 'http://localhost/nagios/'. The Nagios Core interface is displayed, featuring a sidebar with navigation options like 'Network Outages', 'Reports', 'System', and 'Process Info'. The main content area is divided into two panels:

- Process Information:**
 - Program Version: 3.3.1
 - Program Start Time: 11-18-2011 19:23:41
 - Total Running Time: 0d 0h 9m 7s
 - Last External Command Check: 11-18-2011 19:32:40
 - Last Log File Rotation: N/A
 - Nagios PID: 2120
 - Notifications Enabled? **YES**
 - Service Checks Being Executed? **YES**
 - Passive Service Checks Being Accepted? **YES**
 - Host Checks Being Executed? **YES**
 - Passive Host Checks Being Accepted? **YES**
 - Event Handlers Enabled? Yes
- Process Commands:**
 - [Shutdown the Nagios process](#)
 - [Restart the Nagios process](#)
 - [Disable notifications](#)
 - [Stop executing service checks](#)
 - [Stop accepting passive service checks](#)
 - [Stop executing host checks](#)
 - [Stop accepting passive host checks](#)
 - [Disable event handlers](#)
 - [Start obsessing over services](#)
 - [Start obsessing over hosts](#)
 - [Disable flap detection](#)
 - [Enable performance data](#)

The bottom of the window shows a terminal prompt '[root@linux-VM: /ho...]' and a system tray with various icons and the text 'Concluído'.

Figura 33 - Informações dos processos realizados pelo Nagios.

A Figura 34 abrange mais um dos recursos do Nagios, que envolve um registro de checagem sobre o desempenho dos serviços realizados nos últimos 15, 5 ou 1 minutos de processo.

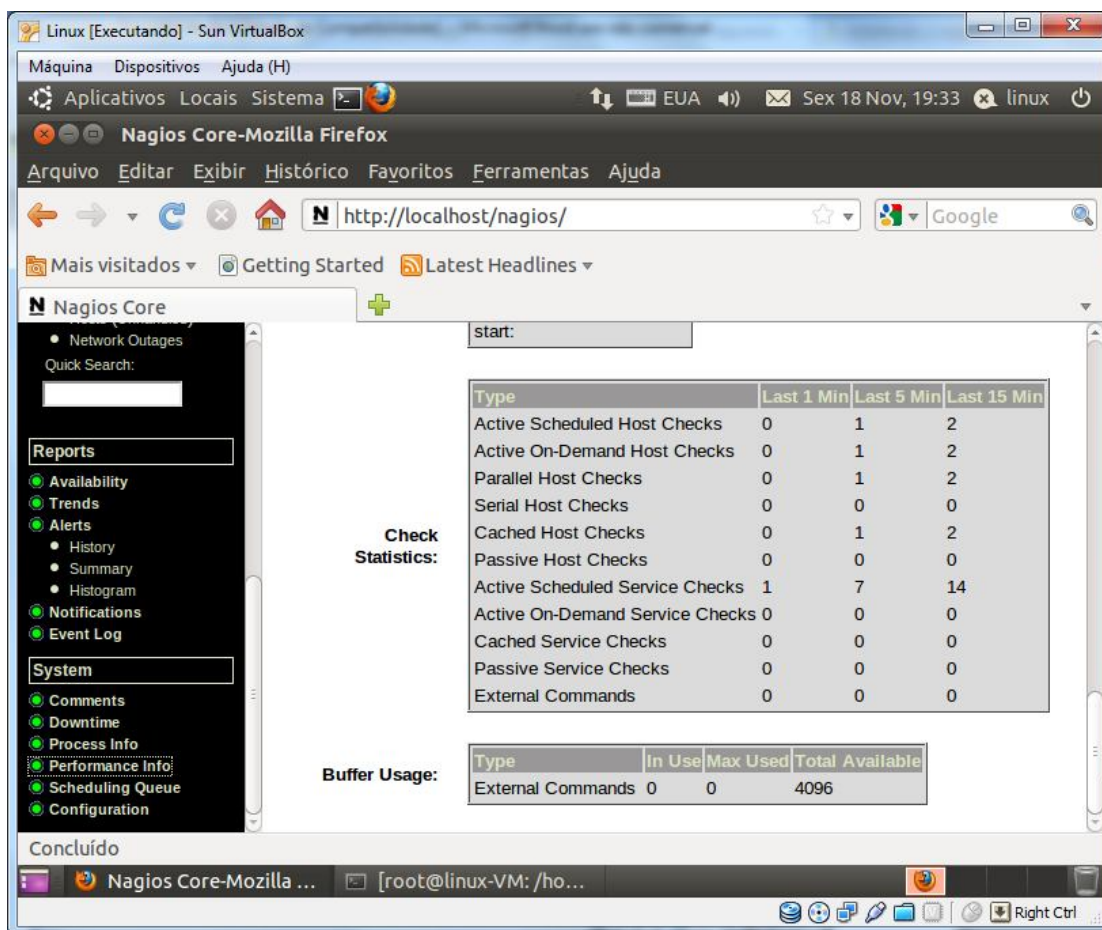


Figura 34 - Checagem de desempenho dos serviços.

Por fim, na Figura 35, um recurso extremamente importante para os administradores que utilizam o Nagios é o sistema de agendamento de serviços a serem executados pelo gerenciador. Nele, o usuário define o tempo em que o programa realizará uma rotina de checagem de determinado processo a fim de se evitar qualquer problema que possa ser ocasionado na rede.

The screenshot shows the Nagios Core web interface in a Mozilla Firefox browser window. The address bar shows the URL `http://localhost/nagios/`. The page title is "Nagios Core". The main content area displays the "Check Scheduling Queue" page, which includes a summary box with the following information:

- Check Scheduling Queue**
- Last Updated: Fri Nov 18 19:33:23 BRST 2011
- Updated every 90 seconds
- Nagios® Core™ 3.3.1 - www.nagios.org
- Logged in as nagiosadmin

Below the summary box, the page shows a table of scheduled checks, sorted by next check time (ascending). The table has the following columns: Host, Service, Last Check, Next Check, Type, Active Checks, and Actions. The data rows are as follows:

Host	Service	Last Check	Next Check	Type	Active Checks	Actions
localhost	Total Processes	11-18-2011 19:28:41	11-18-2011 19:33:41	Normal	ENABLED	[X] [Refresh]
localhost		11-18-2011 19:28:51	11-18-2011 19:34:01	Normal	ENABLED	[X] [Refresh]
localhost	Current Load	11-18-2011 19:29:18	11-18-2011 19:34:18	Normal	ENABLED	[X] [Refresh]
localhost	Current Users	11-18-2011 19:29:56	11-18-2011 19:34:56	Normal	ENABLED	[X] [Refresh]
localhost	HTTP	11-18-2011 19:30:33	11-18-2011 19:35:33	Normal	ENABLED	[X] [Refresh]
localhost	PING	11-18-2011 19:31:11	11-18-2011 19:36:11	Normal	ENABLED	[X] [Refresh]
localhost	Root Partition	11-18-2011 19:31:48	11-18-2011 19:36:48	Normal	ENABLED	[X] [Refresh]

The interface also shows a sidebar with navigation options like "Network Outages", "Reports", "Availability", "Trends", "Alerts", "Notifications", and "System". The status bar at the bottom indicates the user is logged in as nagiosadmin and the system is "Concluído".

Figura 35 - Fila de agendamento de serviços.

7 CONCLUSÃO

Após aplicadas as observações de monitoramento e feitas as análises dos recursos de cada software, as seguintes conclusões foram obtidas:

Em relação ao Cacti, as possibilidades do *software* são muitas, quando utilizado suas funções básicas é possível visualizar gráficos diários, semanais, mensais e anuais sobre utilização de *interfaces* de rede, CPU, memória, espaço em disco, entre outros. Mas quando são adicionadas as funcionalidades desenvolvidas pela comunidade do Cacti, como *plugins* e *templates* diversos este *software* se torna excelente para qualquer área funcional do gerenciamento de redes, além de ficar muito mais robusto e funcional.

A implantação no ambiente de testes se mostrou muito efetiva, tornando-se evidentes as vantagens da implantação do *software* Cacti em qualquer ambiente de rede, devido sua robustez, facilidade de implantação e excelente desempenho, é uma economia para qualquer empresa com suporte de TI, pois, economiza com a aquisição, por ser gratuito, tem aperfeiçoamento constante, com foco na qualidade e diversificação de ferramentas pela comunidade de *software* livre, além de ser possível fazer uma adaptação aos objetivos específicos de cada pessoa ou empresa.

Como ponto fraco, comparado ao Nagios observou-se um desempenho aquém do esperado para levantar, armazenar e exibir os dados, ainda que não seja nada de alarmante.

O *software* Cacti, correspondeu de forma positiva nos testes realizados, demonstrando que esta ferramenta é de extrema importância para garantir um alto nível de confiabilidade e qualidade no gerenciamento de redes em empresas.

Em relação ao Nagios, a instalação e configuração foram trabalhosas, porém a quantidade de listas de discussões na *internet* auxiliou muito seu desenvolvimento. Além disso, o próprio site oficial do Nagios disponibiliza formas de contato bastante ágeis entre os usuários e os desenvolvedores do sistema.

O presente estudo permitiu através do *software*, a avaliação de diversos aspectos da gestão e monitoramento de redes de computadores.

Para o monitoramento de serviços o Nagios se mostrou muito bem aplicável, pois através do uso de seus recursos é possível ter uma visão global da rede. Ele é

um software abrangente e experiente que reporta e atualiza corretamente todas as informações relevantes, dando-se ênfase maior em cima do quesito disponibilidade, tendo este produto diversas ferramentas para monitorar os mais variados serviços e plataformas Windows/Linux/Unix.

Alguns itens podem ser implementados para aprimorar o Nagios: desenvolvimento de um *front-end* de configuração, com o intuito de facilitar e concentrar o meio de configuração dos seus arquivos *cfg*; ampliação da capacidade de monitoramento do Nagios, para abranger arquivos de servidor *web*, evitando, assim, a ação de *hackers*; desenvolvimento de *plugins* com objetivos específicos, voltados para o monitoramento de algum equipamento em particular, como controladores de temperatura, umidade, volume de água, etc.

Com os recursos humanos tornando-se cada vez mais escassos, nenhum departamento de TI pode se dar ao luxo de ter seus sistemas manualmente verificados. Redes estão se tornando mais complexas e demandam especialmente a necessidade de serem informadas o quanto antes, sobre quedas que aconteceram ou por problemas que estão por acontecer.

O Nagios, uma ferramenta de código aberto para monitoração de sistemas e redes, ajuda o administrador a detectar problemas antes que o pior possa acontecer. Devido à eficiência da monitoração, ele não sobrecarrega o servidor nem os dispositivos de rede; permite que outras aplicações possam compartilhar o dado SNMP; o teste de dispositivos é feito de forma rápida; gera relatórios identificando imprecisão na monitoração existente e possui dados de configuração unificados.

O Nagios provê uma visão do essencial de desempenho e disponibilidade, e é um exemplo de como a Comunidade de Código Aberto pode ajudar no gerenciamento da rede.

As ferramentas estudadas apresentam muitas semelhanças entre si e, em geral, fornecem soluções para a maioria das necessidades que o gerenciamento de redes exige. Ainda assim, observam-se algumas características diferenciadas entre elas conforme demonstrado no quadro comparativo a seguir:

QUADRO COMPARATIVO	CACTI	NAGIOS
Agente	Não	Sim
Scripts externos	Sim	Sim
Plugins	Sim	Sim

Linguagem de programação	PHP	Perl
Alertas	Sim	Sim
Front-end Web	Controle Completo	Controle Parcial
Monitoramento distribuído	Sim	Sim
Armazenamento de dados	RRDTool, MySQL	MySQL, MSSQL
Eventos	Através de plugin	Sim
Syslog	Não	Através de plugin

Por fim, os dois softwares analisados neste trabalho mostraram possuir muitas opções para um monitoramento de rede eficaz, sendo em sua essência semelhantes entre si, porém cada qual com suas particularidades que fazem deles ótimos produtos de escolha para um gerenciamento satisfatório e funcional, sem contar o fato de serem totalmente gratuitos.

Espera-se assim que o estudo e análise feitos neste trabalho possam contribuir para que administradores de redes conheçam e desenvolvam melhorias para tais softwares a fim de se melhorar cada vez mais um recurso tão importante no que tange a tecnologia, o gerenciamento de uma rede.

REFERÊNCIAS

ALBUQUERQUE, Fernando. **TCP-IP Internet: protocolos & tecnologias**. 3. ed. Rio de Janeiro : Axcel Books do Brasil, 2001. xv, 362 p.

BLACK, Tomas Lovis. **Comparação de Ferramentas de Gerenciamento de Redes**. Porto Alegre, 2008.

CAMPOS, Augusto. **O que é software livre**. BR-Linux. Florianópolis, 2006.
Disponível em: <<http://br-linux.org/linux/faq-softwarelivre>>. Acesso em: 2 maio 2011.

CASTALDIN, André Giovanni. **Gerência de Redes – Um Estudo de Caso**. Londrina, 2005. Disponível em: <<http://www2.dc.uel.br/noura/document/down=176>>. Acesso em: 16 maio 2011.

COMER, Douglas E. **Interligação em rede com TCP/IP**. 3. ed. Rio de Janeiro. Ed. Campus, 1999.

COMER, Douglas E. **Redes de Computadores e Internet**. Volume II. Ed. Campus, 1999.

COSTA, Felipe. **Ambiente de Redes Monitorado com Nagios e Cacti**. Rio de Janeiro: Ed. Ciência Moderna Ltda., 2008.

COSTA, Reinaldo Candido da. **Conhecendo o Software Livre**. Minas Gerais: Ed. Horizonte, 2010.

DELFINO, Gardel Moreira. **SNMP - Simple Network Management Protocol**. Rio de Janeiro, 1998.

GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. 5. ed. São Paulo: Atlas, 2008.

GOETEN, Luciano Waltrick. **Protótipo de um software agente SNMP para rede Windows**. Blumenau, 2001.

HARNEDY, Sean. **Total SNMP: Exploring the Simple Network Management Protocol**. 2 ed. Prentice Hall PTR, 1997.

LOPES, Taylor. **Redes: Uma introdução ao Nagios**. São Gonçalo, 2010.

MARTIN-FLATIN, J.P.; ZNATY, S.; HUBAUX, J.P. **A Survey of Distributed Enterprise Network and System Management Paradigms**. Journal of Network and Systems Management, New York, v.7, n.1, p.9-26, Mar. 1999.

MELLO, Jorge Lucas de. **Protótipo de um agente SNMP para uma rede local utilizando a plataforma JDMK**. Blumenau, 2000.

MELO, Tiago Maciel. **Monitoramento de Redes de Médio Porte Utilizando Software Livre**. Palhoça, 2007.

RAHM, Jason. **Graphing your LT5 LTM Environment with Cacti**. 2007.

SANTOS, Adriano Pereira. **Implantação de Software de Gerenciamento de Rede baseado nas plataformas Microsoft e Linux**. São Paulo, 2005.

SCHULZ, Murilo Alexandre. **Protótipo de software de gerência de desempenho de um access point de rede sem fio utilizando o protocolo SNMP**. Blumenau, 2004.

SOARES, Luiz F. G. **Redes de computadores**. 2. ed. Rio de Janeiro. Ed. Campus, 1995.

STALLINGS, W. **SNMP, SNMPv2, SNMPv3 and RMON 1 and 2: the practical guide to network management standards**. 3rd ed. Reding: Addison-Wesley, 1999.

STANGE, Rodrigo. **Ferramenta para Gerenciamento de Falhas em Rede Ethernet Baseada em Protocolo SNMP**. Blumenau, 2008.

SZTAJNBERG, Alexandre. **Gerenciamento de redes – Conceitos básicos sobre os protocolos SNMP e CMIP**. 2. ed. Rio de Janeiro. Ed. Conexão 1996.

TANENBAUM, Andrew S. **Redes de computadores**. Rio de Janeiro: Campus, 2003. 945 p. Tradução de: Computers Networks.