UNIVERSIDADE SAGRADO CORAÇÃO

BRUNO MARCUMINI POLA

GERENCIAMENTO DE REDES COM FERRAMENTAS OPEN-SOURCE UTILIZANDO O PROTOCOLO SNMP

BAURU

2013

BRUNO MARCUMINI POLA

GERENCIAMENTO DE REDES COM FERRAMENTAS OPEN-SOURCE UTILIZANDO O PROTOCOLO SNMP

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

2013

Pola, Bruno Marcumini

P7621g

Gerenciamento de redes com ferramentas open-source utilizando o protocolo SNMP / Bruno Marcumini Pola -- 2013.

71f. : il.

Orientador: Prof. Esp. Henrique Pachioni Martins.

Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade do Sagrado Coração – Bauru – SP.

1. Importância do gerenciamento de redes. 2. SNMP. 3. Gerência de redes. I. Martins, Henrique Pachioni. II. Título.

BRUNO MARCUMINI POLA

GERENCIAMENTO DE REDES COM FERRAMENTAS OPEN-SOURCE UTILIZANDO O PROTOCOLO SNMP

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. André Luiz Ferraz Castro
Universidade Sagrado Coração

_____
Prof. Dr. Elvio Gilberto da Silva
Universidade Sagrado Coração

Bauru 02, de junho de 2013.

Dedico este trabalho aos meus pais, irmã amigos e minha namorada por toda a paciência que tiveram para aguentar meus estresses e por me darem força nos momentos que mais precisei.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por me dar uma cabeça e um corpo capaz de pensar e agir, por me dar capacidade de trabalhar para ajudar a manter o meu curso.

Agradeço aos meus pais, que sempre me apoiaram e me ajudaram na graduação, que mesmo quando estávamos em períodos de turbulência, eles me apontavam os caminhos mais corretos a seguir.

Agradeço a todas as tias (que não são poucas) corujas, que sempre estavam querendo saber as quantas andava minha faculdade e se preocupavam com isso.

Agradeço a minha avó, por a cada dia me ensinar a viver uma vida nova e diferente.

Agradeço a minha irmã, que é meu motivo de orgulho.

Agradeço a todos os primos e primas, por todos os momentos de amor e amizade que sempre passamos juntos.

Agradeço a meu sogro e sogra, que estão sempre se preocupando comigo.

Agradeço a minha namorada, que sempre teve muita paciência para me aguentar e me deu muita força em todos os momentos que precisei.

“Bazinga!”

Sheldon Cooper, PhD

RESUMO

A evolução tecnológica provoca um grande impacto na sociedade atualmente. A informação tem-se tornado objeto de significante 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. Uma rede mal montada é sinônimo de prejuízo para qualquer empresa. 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. Diante desses fatos essa pesquisa tem como objetivo mostrar a importância de um ambiente de rede gerenciado e as melhorias que se pode ter com o inicio do gerenciamento.

Palavras-chave: Importância do gerenciamento de redes, SNMP, gerencia de redes.

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 up. A network poorly assembled is synonymous of prejudice to any company. To maintain control over the situation emerged the concept of management of networks that seeks to maximize efficiency and productivity over the information that travels over the network. Faced with such facts these research aims to show the importance of a network environment managed and improvements that may have when starting the management.

Key-words: Importance of maneged networks, SNMP, Network Management.

LISTA DE ILUSTRAÇÕES

LISTA DE FIGURAS

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

A expansão das redes de computadores e o surgimento de novas tecnologias estão crescendo a cada dia. Atualmente, as redes e os recursos associados a elas são fundamentais e de extrema importância para uma organização. É imprescindível que elas não falhem e que os tempos de indisponibilidade sejam minimizados.

Tais redes fornecem suporte a uma variedade de atividades humanas como, por exemplo, a maneira com que se aprende ou se informa. Através de ferramentas como mensageiros instantâneos, blogs, podcasts e wikis a informação é compartilhada de forma rápida e de maneira colaborativa, momento em que os usuários da rede participam ativamente deste processo utilizando tais ferramentas para disponibilizar seus próprios conteúdos ao público. Uma grande vantagem nesse caso é a rápida e precisa atualização de conteúdos, o que não acontece com livros, por exemplo. Pode-se também citar outras vantagens como disponibilidade destes recursos a um público muito maior, redução de custo e consistência na qualidade de ensino. Redes de computadores também influenciam a maneira de se trabalhar e se divertir. Muitas companhias disponibilizam uma imensa quantidade de material voltado ao entretenimento como músicas, vídeos, aplicações, games etc. Profissionais podem trabalhar a distância, mesmo de suas residências, o que cria uma nova classe operária, inclusive, os "Teleworkers". Enfim, as redes de computadores evoluíram de tal maneira na atual sociedade que muitas das tecnologias tornaram-se dependentes das tecnologias que envolvem as redes para funcionar. Resumidamente, as redes mudaram, inclusive, a forma de se viver. (DYE, Rick MCDONALD, RUFI, 2008, p.2)

Com isso, a administração e gerência dos recursos de Tecnologia da Informação têm sido uma demanda constante no ambiente corporativo, necessitando então de um gerenciamento eficaz destes recursos. Para comprovar a importância do gerenciamento de ativos de redes, esse trabalho de conclusão de curso apresenta a implementação de um software de gerenciamento em um ambiente que nunca passou por uma avaliação.

Para qualquer que seja a ferramenta utilizada para gerenciar os ativos de redes, é importante saber que essa prática provê aos gestores e administradores de parques computacionais um importante auxílio na tomada de decisões estratégicas sobre a infraestrutura instalada.

Com o veloz crescimento da quantidade de serviços suportado por uma rede de computadores a gerência de redes torna-se tarefa complexa, tanto em desempenho como em suporte. Além disso, os sistemas de telecomunicações, também adicionam complexidade a estas redes e estarão cada vez mais presentes mesmo em pequenas instalações.

A gerência está associada ao controle de atividades e ao gerenciamento do uso dos recursos. Tais tarefas, simplificadamente, são obter informações da rede, tratar estas informações possibilitando um diagnóstico, e encaminhar as soluções dos problemas. Para cumprir estes objetivos, funções de gerência devem ser embutidas nos diversos componentes de uma rede, possibilitando descobrir, prever e reagir a problemas.

2. OBJETIVOS

2.1. OBJETIVO GERAL

Implantar em um ambiente que não tenha nenhuma ferramenta de gerenciamento alguma solução para o gerenciamento, propondo e demonstrando os benefícios de um ambiente gerenciável.

2.2. OBJETIVOS ESPECÍFICOS

- Pesquisar e entender sobre gerenciamento de redes;
- Implantar ferramentas de rede em um ambiente;
- Analisar os relatórios gerados por um software de gerenciamento de rede livre;

3. JUSTIFICATIVA

Com o advento da evolução tecnológica, o diferencial que mais impactou os negócios da sociedade foram as redes de computadores. Tudo isso foi impulsionado graças ao avanço das telecomunicações. Assim as redes de telecomunicações deixaram de ser luxo e passaram a ser uma importante estratégia, hoje uma necessidade não só para as grandes empresas, como para qualquer pessoa comum.

Na década de 50, onde surgiram os primeiros sistemas de computadores baseados em sistema de informação, as necessidades computacionais das empresas foram substituídas de um único equipamento atendendo todas as necessidades para um conjunto de máquinas autônomas interconectadas, que podem trocar informações. Assim, o uso da rede para compartilhamento de recursos passou a ser indispensável no cotidiano de uma empresa e também em ambientes domésticos.

A rede de computadores não se limita a internet, ela está presente em varias atividades do cotidiano, como em serviços bancários, chamadas telefônicas, entre outros. O que podemos perceber é que estamos cada dia mais dependentes desses serviços e por consequência, na utilização das redes. Nesse contexto, o uso de um computador que não esteja conectado a uma rede esta ficando sem utilidade, a facilidade e comodidade que são adquiridas utilizando esses serviços faz com que haja dependência. Se o serviço de rede de uma empresa estiver inativo todos os departamentos são afetados. O que leva ao desenvolvimento desse Trabalho de Conclusão de Curso, a pesquisa e instalação de um software de gerenciamento, a fim de obter dados que comprovem a importância de gerenciar os ativos de rede de computadores.

4. REVISÃO DA LITERATURA

4.1 GERENCIAMENTO DE REDE

"O gerenciamento de rede é o procedimento que consiste em controlar todos os componentes de hardware e software da rede." (RIGNEY, 1996, p.148).

De acordo com Albuquerque (2001) as redes prestam serviços essenciais em todas as organizações e à medida que as redes locais ficam maiores e interligadas com outras redes, é necessário um sistema que facilite essa gerencia.

Uma característica essencial ao administrador ou gerente de uma rede é que o responsável tenha amplos conhecimentos de procedimentos, desempenho e identificação de falhas que possam acontecer e familiarização com os sistemas por ele utilizados no cotidiano. Por isso é importante para um gerente de rede conhecer informações sobre os componentes da rede, como tipo de processador, quantidade de memória de cada computador, sistema operacional instalado, quantidade de Switches, roteadores e vários outros.

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 Tanembaum (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 de redes é tarefa extremamente importante 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 gerenciar e controlar o 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, consequentemente, 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 seu 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, consequentemente, 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 gerenciada 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:

4.1.3.1 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.

4.1.3.2 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 mal 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.

4.1.3.3 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.

4.1.3.4 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 essências 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.

4.1.3.5 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.

4.2 HISTORICO DO GERENCIAMENTO DE REDES

Desde 1986 o comitê técnico em comunicação de dados International Federation for Information Procesing (IFIP) havia o consenso sobre a necessidade de gerenciamento. Os representantes incorporavam apenas as três camadas inferiores da arquitetura Open Systems Interconnection (OSI), 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 disponibiliza-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) relata que a abordagem para integrar o gerenciamento de redes era 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. Um dos maiores problemas dessa solução é a limitação imposta pelo fato de somente usar opções de gerenciamento que tinham semelhança com a arquitetura proprietária do fornecedor do computador gerenciador. Opções de interação que estavam disponíveis pelos dispositivos gerenciados podiam não ser aproveitadas simplesmente pela falta de condições de mapeá-las para uma forma que o computador gerenciador reconhecesse. Em decorrência, os dispositivos gerenciados providos pelo mesmo fornecedor do computador gerenciador apareciam mais facilmente.

De acordo com Black (2008), o primeiro dos protocolos de gerência de rede foi o Simple Gateway Monitoring Protocol (SGMP) que surgiu em novembro 1987. Porém, o SGMP era restrito à gerenciamento 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.

Em 1988, houve uma revisão de protocolos pela Internet Architecture Board (IAB) e 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.

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.

4.3 PROTOCOLO SNMP

Este protocolo, que opera na camada de aplicação, como mostra a Figura 01, pode ser facilmente implementado e consome poucos recursos das máquinas e dos canais de comunicação. O código necessário à sua implementação pode ser desenvolvido para dispositivos com capacidades mínimas de processamento e armazenamento, e a sobrecarga decorrente do uso do SNMP na rede e nas entidades é pequena (ALBUQUERQUE, 2001).

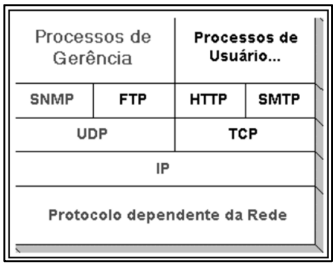

Figura 1 - Protocolo de Gerenciamento SMNP.
Fonte – MELLO (2000)

Atualmente existem três versões de SNMP: o SNMPv1, o SNMPv2 e o SNMPv3. O SNMPv3, implementa as questões de segurança não encontradas nas primeiras versões do SNMP, além de adicionar novas funcionalidades, onde destacam-se a capacidade de gerenciamento distribuído, via primitivas

de comunicação gerente-gerente, e as formas de tratamento e transporte de dados (GOETEN, 2001).

Na arquitetura SNMP, a maior parte do processamento ocorre nas estações de gerência e não nas entidades gerenciadas. Essa arquitetura é composta por agentes, estações de gerência, bases de dados com informações necessárias à gerência e protocolo para a comunicação entre agentes e estações. O protocolo define as estruturas das mensagens e a sequência em que devem ser trocadas informações entre agentes e estações de gerência. São definidas mensagens para a leitura de informações dos agentes, a escrita de informações nos agentes e os eventos notificados pelos agentes. Os formatos destas mensagens foram estabelecidos através de uma linguagem formal chamada Abstract Syntax Notation One (ASN.1) (ALBUQUERQUE, 2001).

O banco de dados que modela o agente SNMP é denominado Management Information Base (MIB) e sua função básica é estabelecer quais valores podem ser gerenciados no dispositivo. O SNMP permite a extensão destes valores padrões adicionalmente com valores específicos para um agente particular pelo uso de MIB privados. Diretivas emitidas pelo gerenciador da rede a um agente SNMP consistem nos identificadores de variáveis de SNMP (chamados identificadores da MIB ou variáveis da MIB) junto com instruções para adquirir o valor do identificador ou fixar o identificador para um novo valor (MELLO, 2000).

4.4 GERENTE E AGENTE

Comer (1999, p. 437) cita que o Agente é um processo executado em uma máquina gerenciada, que é responsável pelas informações de gerência da máquina. Ele tem duas funções principais: atender as requisições enviadas pelo gerente e enviar automaticamente informações de gerenciamento quando programado.

Comer (1999, p. 437) ainda diz que o gerente é um programa executado trabalhando em um servidor, permitindo a obtenção e envio dos dados de gerenciamento junto aos dispositivos gerenciados mediante a comunicação com um ou mais agentes. Ele é responsável pelo gerenciamento, 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, 1997, p. 419).*

Figura 2 - Componentes do SNMP.
Fonte – Harnedy, 1997

4.5 MANEGEMENT INFORMATION BASE (MIB)

A MIB é o conjunto dos objetos gerenciados que traz todas as informações necessárias para o gerenciamento. Ela contém uma lista de variáveis, denominadas de objetos, e seus respectivos atributos. O principal requisito para o correto funcionamento do gerenciamento é a estrutura da MIB contemplar os recursos disponíveis do equipamento gerenciado e estes recursos poderem ser lidos pelo gerente de rede. Por este motivo surgiu a necessidade de padronizar uma lista de objetos, que deu origem a MIB.

Historicamente a MIB foi desenvolvida em duas etapas. A primeira delas foi apresentada pela RFC (*Request for Comments*) 1156 trazendo a MIB primeira versão. Logo após foram contempladas algumas melhorias na estrutura desta MIB que deram origem a MIB-2, através da RFC 1213, utilizada atualmente.

O protocolo SNMP utiliza uma MIB padrão, que é conhecida como *Structure of Management Information* (SMI). Nessa estrutura, somente cinco tipos de dados são permitidos: *integer, bit string, octet string, null e object identifier*. A partir destes tipos primitivos citados acima, podem ser construídos

objetos mais complexos. A variável *Object Identifier* oferece uma forma de identificar objetos. O mecanismo utilizado é definir uma árvore de padrões e colocar todos os objetos de cada padrão em um único local na árvore. Na figura 3 pode-se ver parte da árvore que inclui a MIB do SNMP (SCHULZ, 2004).

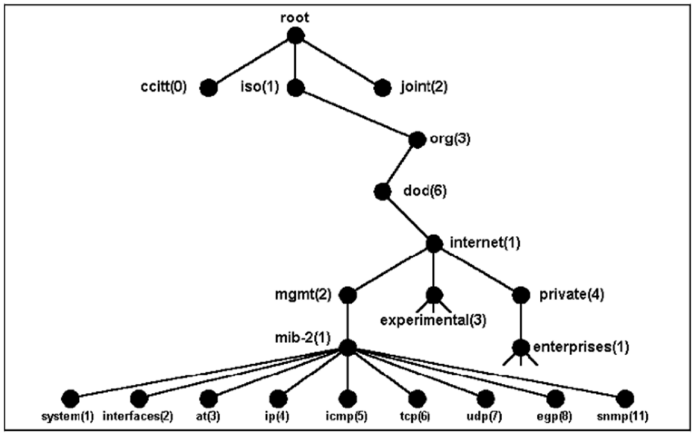

Figura 3 - SMI: MIB padrão SNMP
Fonte – Schulz (2004)

Segundo Schulz (2004), o nó raiz da árvore não possui rótulo, mas possui pelo menos três sub níveis, sendo eles: o nó 0 que é administrado pela *Consultative Committe for* 28 *International Telegraph and Thelephone* (CCITT), o nó 1 que é administrado pela ISO e o nó 2 que é administrado em conjunto pela CCITT e pela ISO. Sob o nó ISO fica o nó que pode ser utilizado por outras instituições: o org (3). Abaixo dele fica o DOD (6) que pertence ao Departamento de Defesa dos EUA. O DOD definiu seis árvores, na qual um sub-nó para a comunidade Internet, que é administrado pela *International Activities Board* (IAB). Abaixo desse nó tem-se:

- directory (1): mantém informações sobre o X.500, serviço de diretórios da ISO;
- mgmt (2): contém as informações de gerenciamento. É nesta árvore que fica o nó da MIB-2 da internet;
- experimental (3): contém projetos experimentais da IAB;
- private (4): contém objetos definidos por organizações privadas;

- security (5): objetos definidos especificamente para assuntos de segurança;
- SNMPv2 (6): objetos definidos especificamente para o SNMPv2.

A análise da estrutura acima revela a origem do *Object Identifier*. Todos os nós possuem além do nome um número particular. Este número é utilizado em sequências para identificar os objetos de interesse. Dessa forma, os objetos da MIB SMI são sempre identificados com o prefixo 1.3.6.1.2.1, que resulta da sequência iso, org, dod, internet, mgmt, mib-2. Após este prefixo surgem as categorias, que identificam os objetos de cada equipamento gerenciado. Estas categorias estão listadas no Quadro 1:

Categoria	N. de Objetos	Informação
System	7	Nome, localização e descrição do equipamento
Interfaces	23	Interfaces de rede e dados de tráfego
Addr-Translation	3	Tradução de endereços
IP	42	Estatísticas de pacotes IP
ICMP	26	Estatísticas de mensagens ICMP recebidas
TCP	19	Algoritmos TCP, parâmetros e estatísticas.
UDP	6	Estatísticas de tráfego UDP
EGP	20	Estatísticas de tráfego do protocolo do Gate-way exterior
Transmission	0	Reservado para MIBs específicas para mídia
SNMP	29	Estatísticas de tráfego SNMP

Figura 1- Objetos da MIB-2.
Fonte – Schulz (2004)

O grupo *System* da MIB-2 contém informações como nome do dispositivo, tipo de equipamento, fabricante, modelo, data de última inicialização. O grupo *Interface* trata dos adaptadores de rede, controlando o número de pacotes e bytes enviados e recebidos da rede, descartes, difusões e tamanho da fila. O grupo *Addr-Translation* fornece informações sobre o mapeamento de endereços. O grupo *IP* trata de todo o tráfego *IP* recebido e transmitido pelo equipamento. São especialmente importantes para o

gerenciamento de roteadores. O grupo *ICMP* se refere a mensagens de erro *ICMP* registrando quantas mensagens de erro foram encontradas. O grupo *TCP* monitora conexões abertas, segmentos enviados e recebidos e erros. O grupo *UDP* registra o número de datagramas *UDP* enviados e recebidos e estatísticas de erros. O grupo *EGP* é usado para controlar roteadores compatíveis com este protocolo. O grupo *Transmission* é um marcador de lugar para MIBs de meios físicos externos. Por exemplo, é possível manter estatísticas especificamente relacionadas a Ethernet. O grupo *SNMP* se destina ao cálculo de estatísticas sobre a operação do próprio *SNMP* (SCHULZ, 2004).

A definição dos objetos (variáveis) numa MIB do SNMP padrão, segundo Goeten (2001), contém os seguintes tipos de dados:

- ***INTEGER, OCTET, STRING, NULL, OBJECT IDENTIFIER, SET e SEQUENCE*** – são tipos universais, de uso geral;
- ***IpAddress*** – um endereço de 32 bits utilizando o formato IP;
- ***Counter32*** – um inteiro positivo que pode ser incrementado, mas nunca decrementado. Seu valor máximo é 2^{32} – 1(4.294.967.295). Quando atingir seu valor máximo é reiniciado em zero;
- ***Gauge32*** – um inteiro positivo que pode ser incrementado e decrementado. Seu valor máximo é o mesmo do Counter32. Quando este valor máximo é alcançado ele não é reiniciado, pois pode ser decrementado;
- ***TimeTicks*** – este inteiro positivo conta, em milésimos de segundos, um determinado período;
- ***Opaque*** – este tipo permite suportar dados arbitrários. O dado é codificado como um OCTET STRING para transmissão.

Os objetos apresentados na MIB são classificados basicamente em três grupos: *read-only* (apenas leitura), *write-only* (apenas escrita), *read-write* (leitura e escrita). Esta classificação é baseada na forma como o objeto pode ser acessado ou alterado. Um objeto *read-only* pode apenas ser lido, sem direito a alteração. Objetos com direito a escrita podem ter seus valores alterados por meio de comandos SET. Para permitir qualquer tipo de acesso

aos objetos foram atribuídas duas classes de senhas. Estas senhas permitem o acesso de leitura ou acesso de escrita nos valores da MIB, também são chamadas de *communities* (comunidades). Através destas senhas é possível restringir a interação do gerente com os objetos gerenciados. Normalmente os acessos de leitura são liberados para que os usuários e gerentes com pouca experiência possam consultar informações sobre o equipamento. O acesso de escrita nos valores deve ser controlado pelos gerentes de nível superior, pois estes permitem alteração nas configurações de equipamentos e objetos.

4.6 FUNCIONAMENTO DO SNMP

O protocolo SNMP foi desenvolvido para rodar sobre a camada de transporte. 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).

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 que não comprometa o 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 o desempenho e as informações.

Como o UDP é um protocolo não confiável, é possível que mensagens SNMP sejam perdidas. As ações a serem tomadas quando da perda de uma mensagem SNMP não são abordadas pelo padrão (MELLO, 2000). 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.

São cinco comandos básicos que o SNMP utiliza para suas operações:

a) *Get-Request* solicita que os nomes das variáveis requeridos sejam informados ao gerente;
b) *Get-Next-Request* solicita a variável seguinte, permitindo que um gerente percorra a MIB inteira;
c) *Get-Bulk-Request* serve para a transferência de grandes quantidades de informação, como por exemplo uma tabela de dados;
d) *Set-Request* permite atualizar o valor de uma variável, mudando o estado desta, desde que a especificação do objeto permita essas atualizações;
e) *Inform-request* tem a utilidade de informar a um gerente quais as variáveis ele está gerenciando;
f) *Trap* é uma mensagem enviada de um agente para um gerente quando acionada;

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

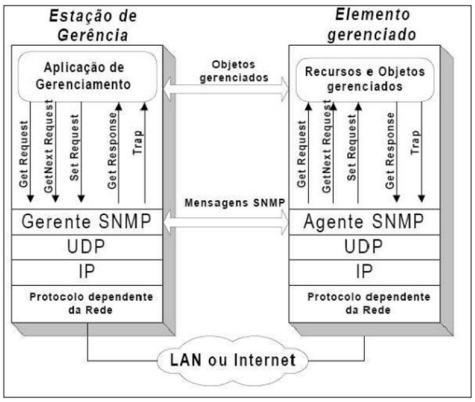

Figura 4 - UDP como Protocolo de Transporte.
Fonte – Mello (2000)

A formação das mensagens SNMP é feita, diferente da maioria dos protocolos. Primeiramente é formado o pacote com todas 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).

4.7 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.

4.8 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. Gerencia 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 RDDTool. (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 gerenciar 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 gerenciados, 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 5, observa-se o gerenciamento de alguns serviços pelo Cacti:

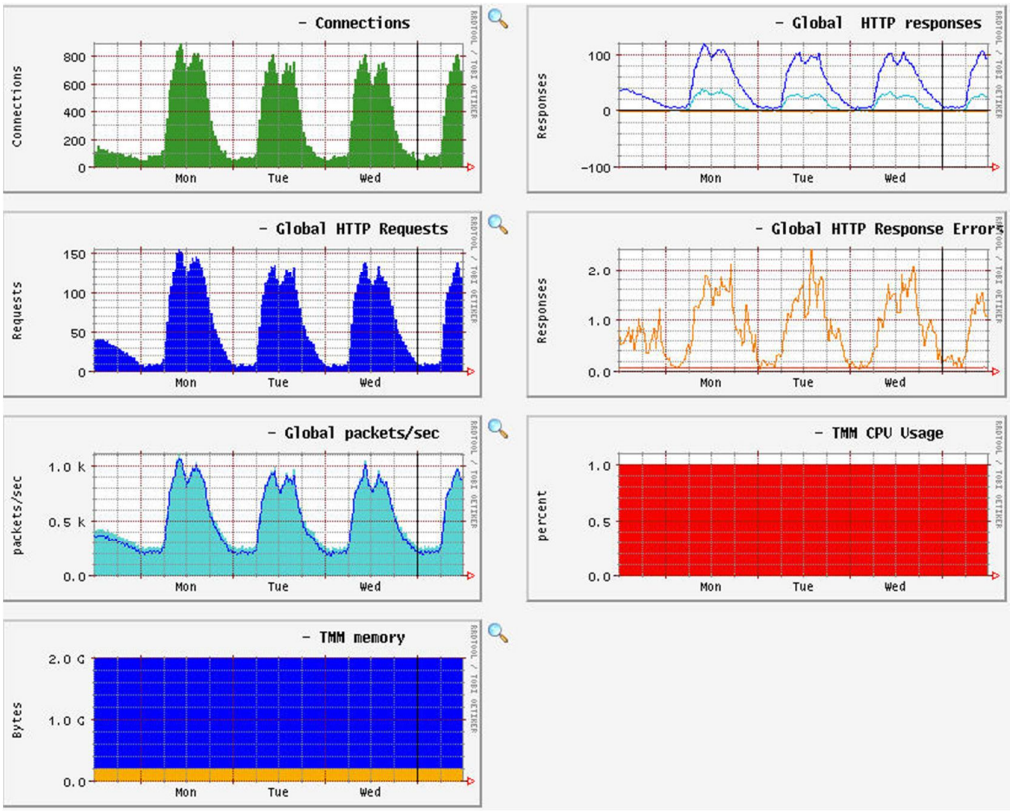

Figura 5 - Gerenciamento de alguns serviços pelo Cacti

Fonte: Rahm (2007).

4.9 NAGIOS

De acordo com Lopes (2010), o Nagios é um aplicativo de gerenciamento 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:

- Gerenciamento de rede e serviços;
- Gerenciamento 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 gerenciamento 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:

- Um servidor *web* (preferencialmente Apache);
- 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 gerenciamento da rede.

Na Figura 6, observa-se o gerenciamento de alguns serviços pelo Nagios:

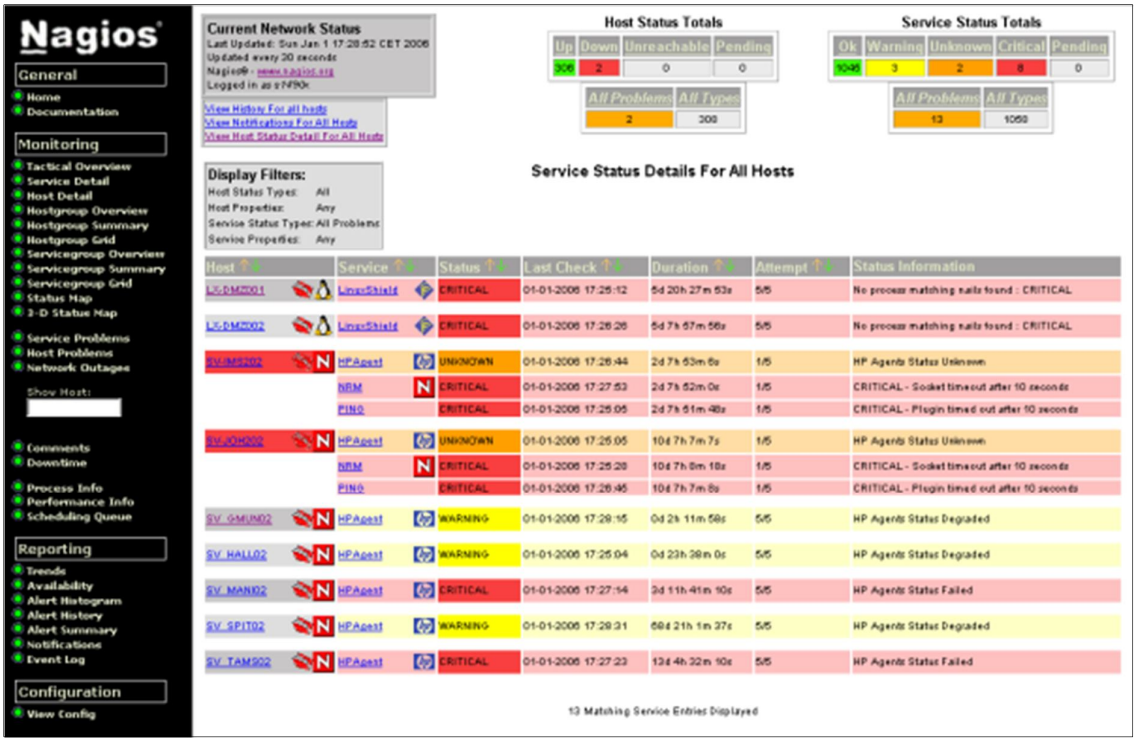

Figura 6 - Gerenciamento de alguns serviços pelo Nagios

Fonte: Lopes (2010).

5. METODOLOGIA

5.1 TIPO DE PESQUISA

Está é uma pesquisa exploratória, que visa proporcionar maior familiaridade com o problema pesquisado, com vistas e torna-lo mais explícito, aprimorar ideias e descobrir intuições. Busca também estender o estudo do tema à integração com o problema, de maneira a construir hipóteses ou apenas explicitar os resultados. Envolve levantamento bibliográfico, parecer das pessoas que possuem experiências práticas com o problema pesquisado e análise de exemplos que incitem a compreensão (GIL, 2008).

5.2 MATERIAIS

O desenvolvimento do trabalho se deu em um microcomputador pessoal (notebook) com o sistema operacional Windows 8, equipado com um processador Intel Core i3 M330 de 2.13GHz, 4 Gb de memória RAM, 500 Gb de Disco Rígido, sendo ele emulado em uma máquina virtual com o sistema operacional Linux Ubuntu 13, 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 gerenciamento 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 PRODEDIMENTOS

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 Oracle VM Virtualbox no computador pessoal. O sistema operacional usado na máquina virtual foi o Linux Ubuntu versão 13.

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 13. O programa utilizado para emular a máquina virtual foi o Oracle VM Virtualbox.

Depois de 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 roda o Cacti e seus plugins, devido eles serem feitos em php.

 # apt-get install php5

Instalação do Php 5, Ele é necessário para poder rodar 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 7.

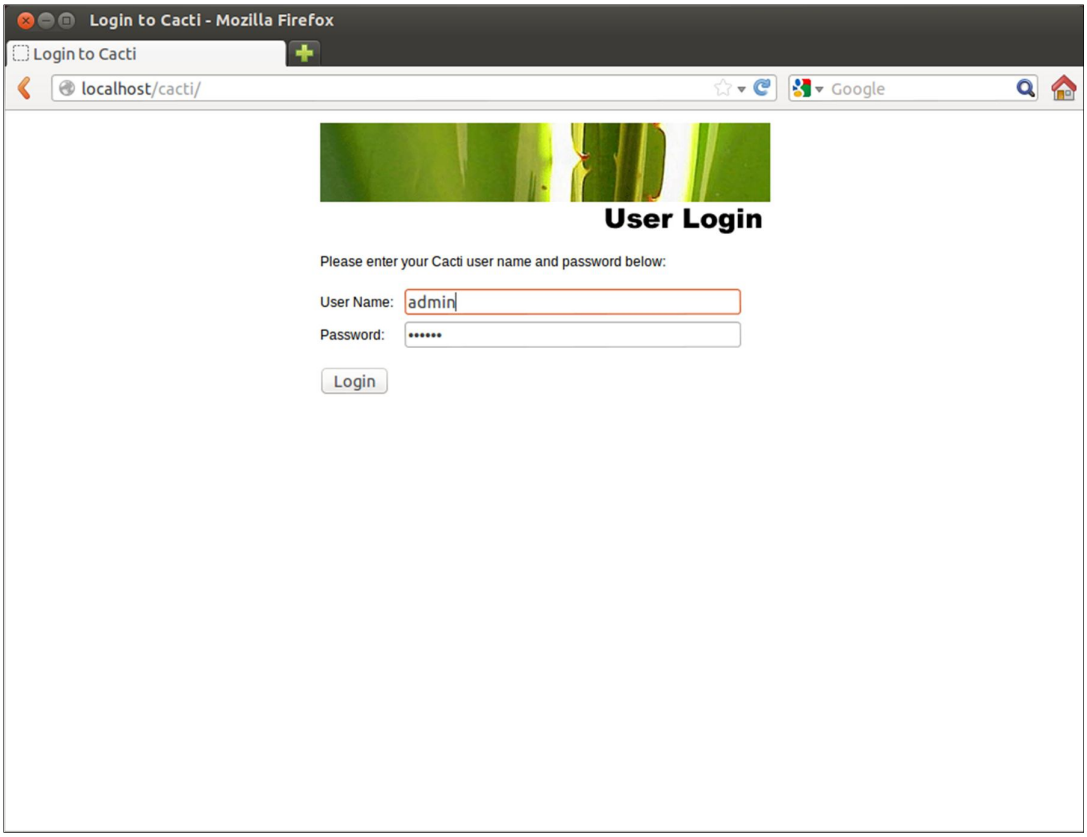

Figura 7 - Tela de Login do Cacti
Fonte: The Cacti Group, Inc.

Na figura 8, é possível vizualizar a tela inicial do Cacti onde se podem acessar suas abas principais de recursos e adicionar dispositivos para gerenciar. Juntamente com esses dispositivos podem ser criados gráficos para vizualização de estatísticas sobre o gerenciamento.

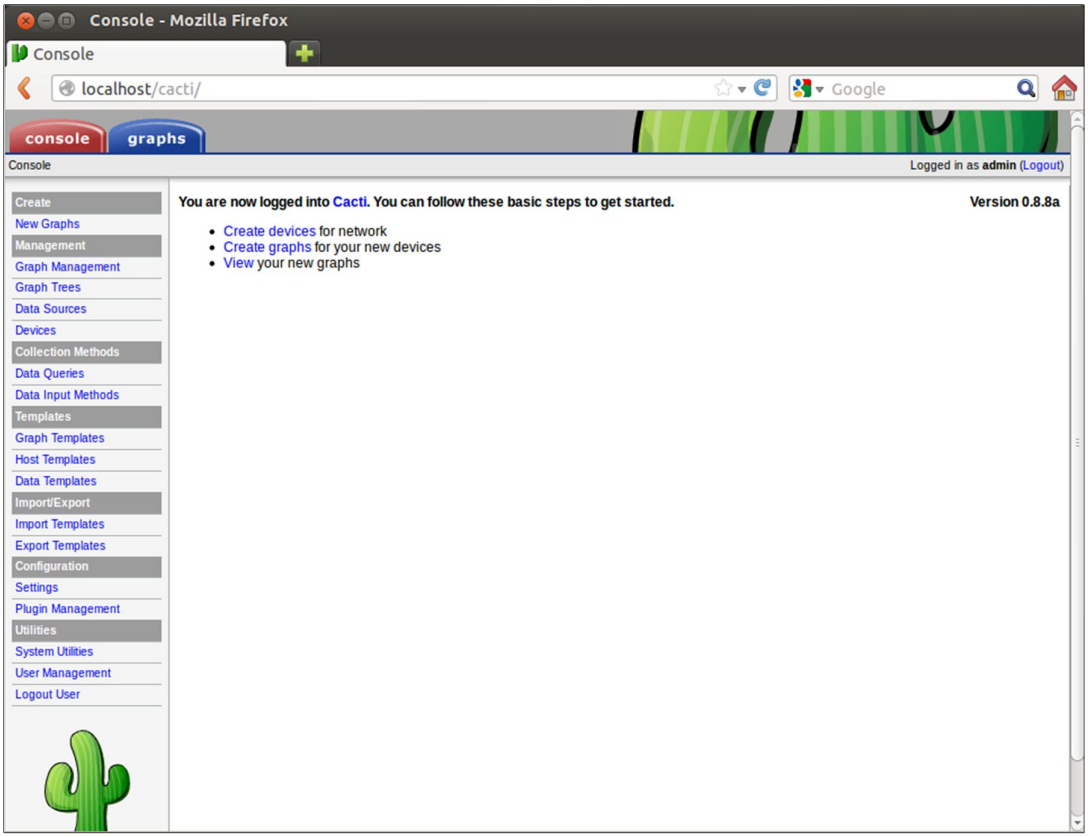

Figura 8 - Tela inicial do Cacti
Fonte: The Cacti Group, Inc.

Já na Figura 9, uma das funcionalidades do Cacti, o uso de memória, é mostrado graficamente. Ele possui um sistema de gerenciamento com marcações legendadas do dia e das horas gerenciadas, além de detalhar a quantidade em kilobytes usada pela memória.

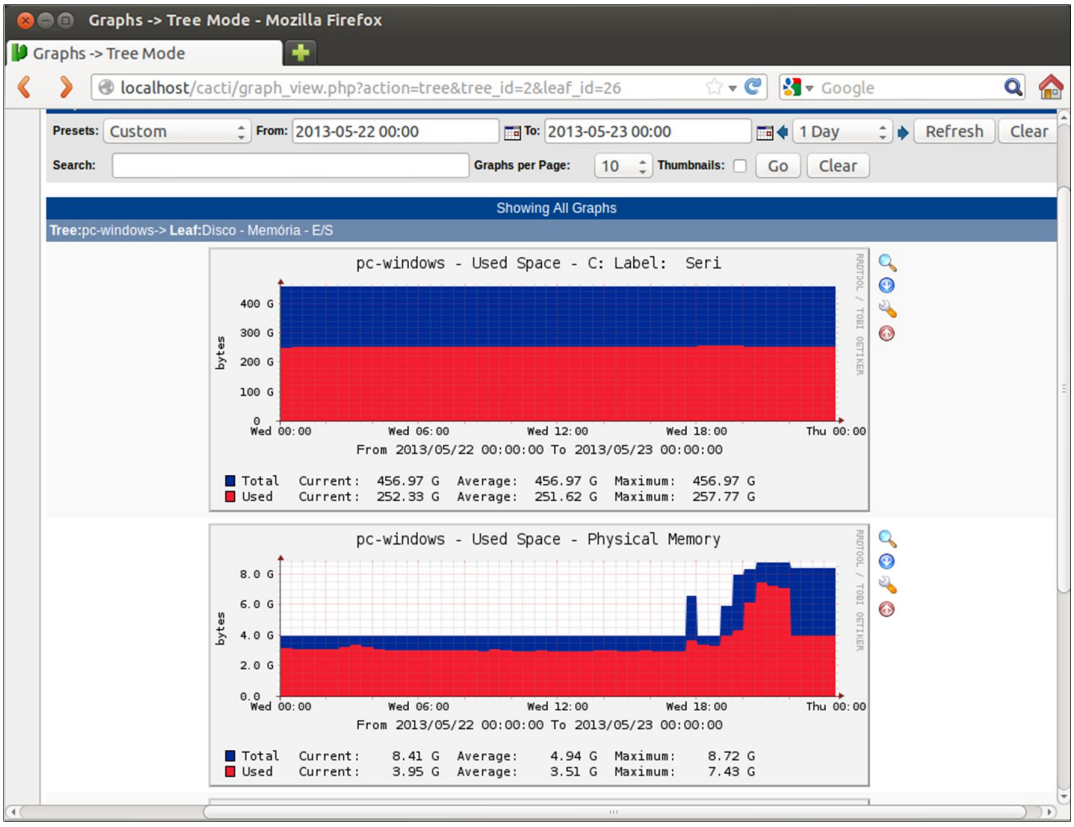

Figura 9 - Gráficos de uso de memória no Cacti
Fonte: The Cacti Group, Inc.

Nas figuras 10 e 11 outra das funcionalidades do Cacti, é mostrada graficamente a utilização da CPU.

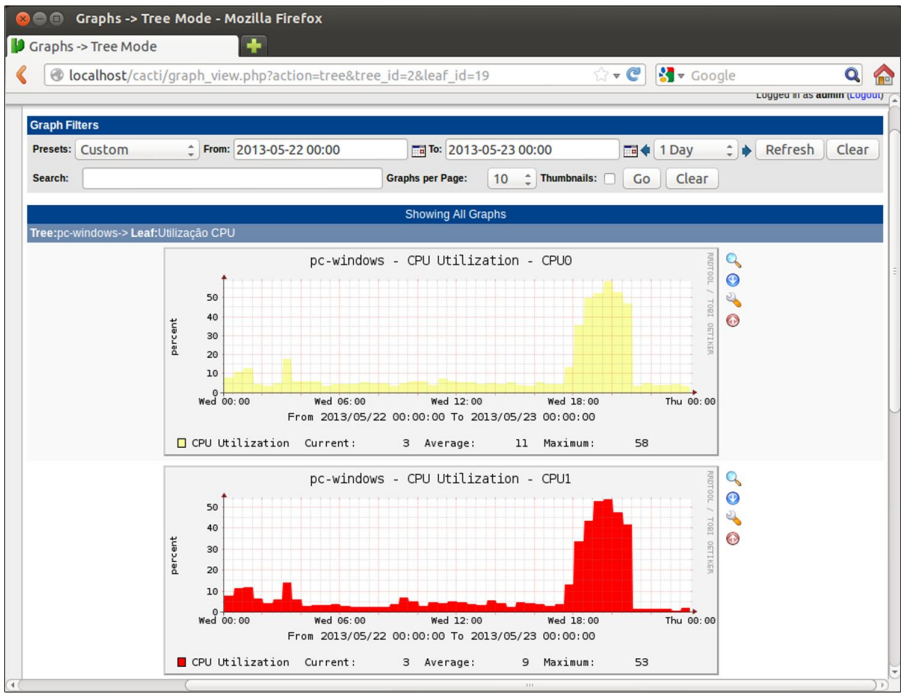

Figura 10 - Utilização da CPU no Cacti
Fonte: The Cacti Group, Inc.

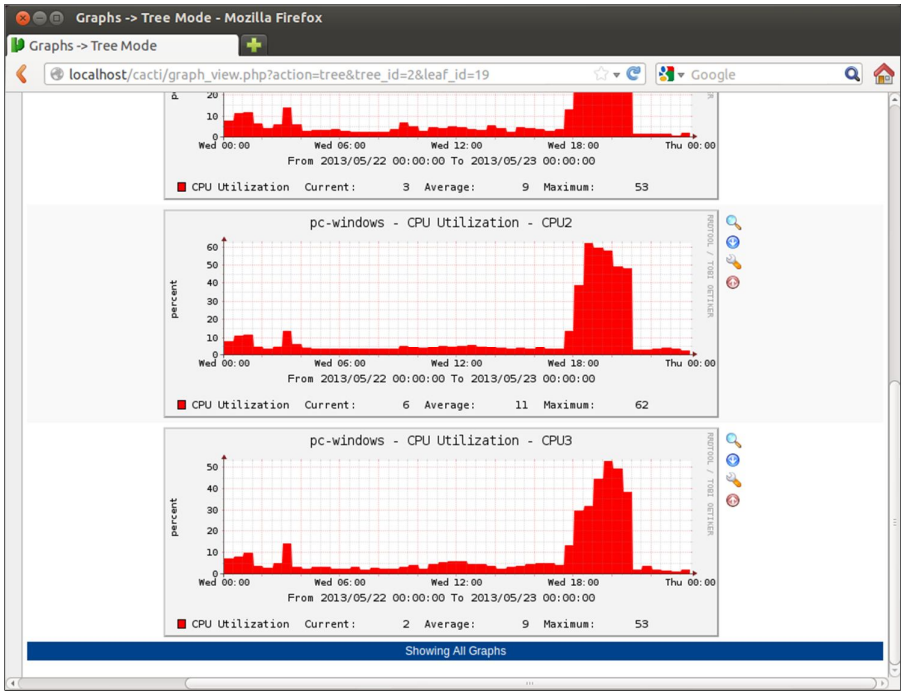

Figura 11 - Utilização da CPU no Cacti
Fonte: The Cacti Group, Inc.

Na Figura 12, mais uma das funcionalidades do Cacti, tráfego de rede é mostrado graficamente. Nele, são mostrados a quantidade de tráfego gerado na rede por um sistema de gerenciamento com marcações legendadas do dia e das horas gerenciadas.

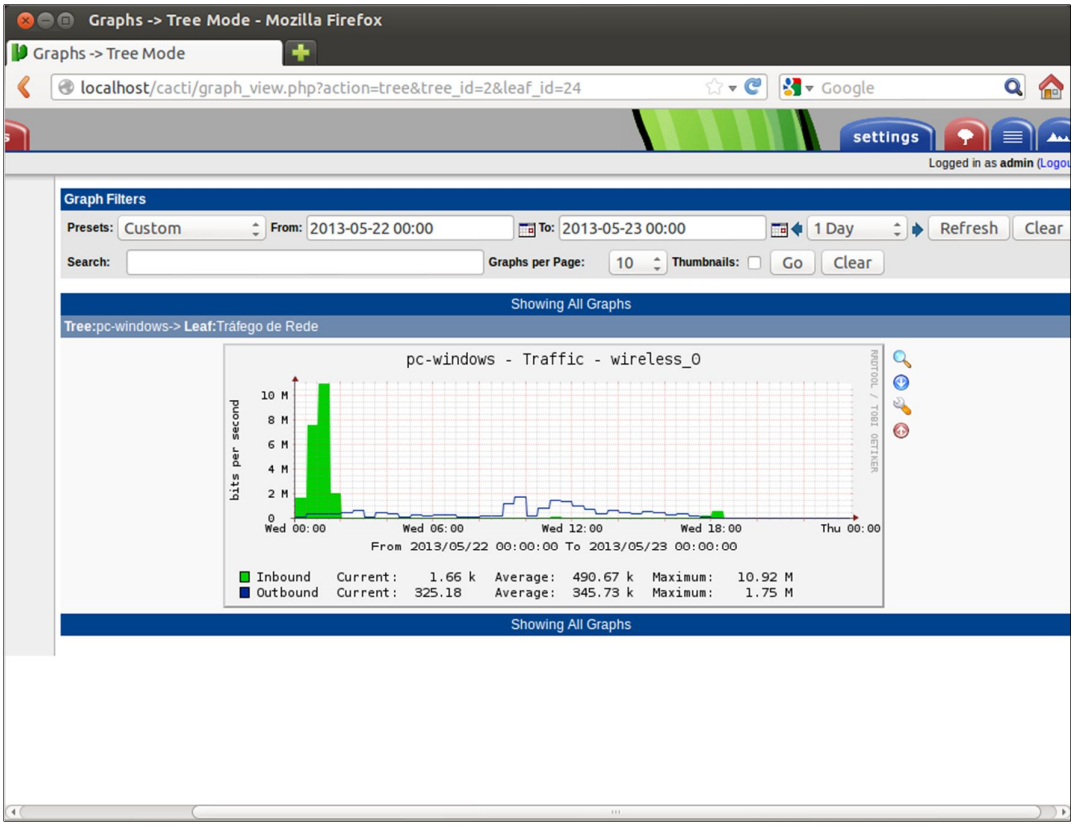

Figura 12 - Tráfego de rede no Cacti
Fonte: The Cacti Group, Inc.

Todos esses gráficos e dispositivos mostrados anteriormente podem ser configurados de acordo com os recursos disponíveis para o usuário conforme mostram as Figuras 13 e 14.

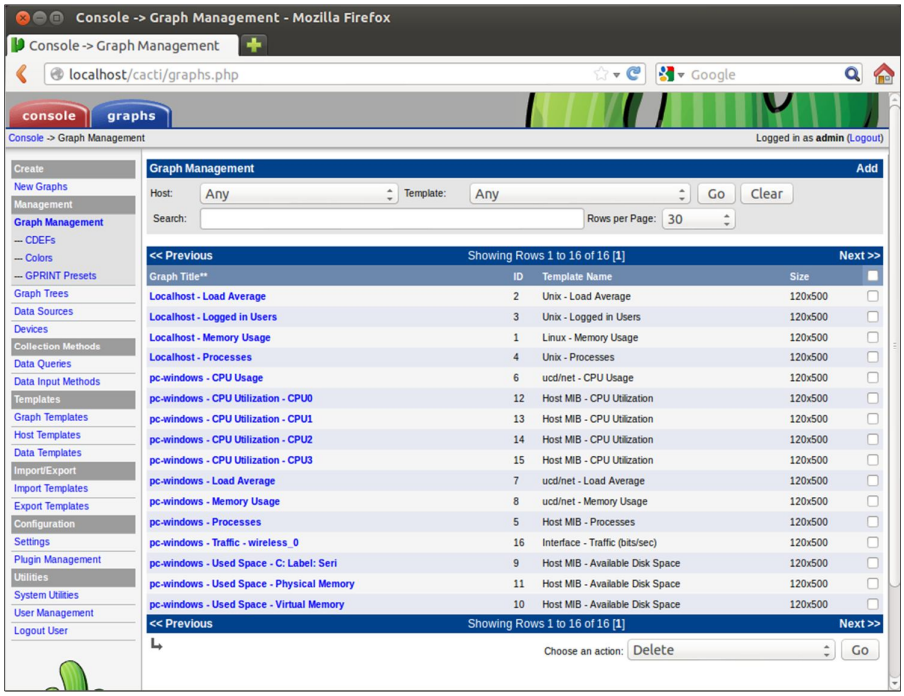

Figura 13 - Gerenciamento de gráficos no Cacti
Fonte: The Cacti Group, Inc.

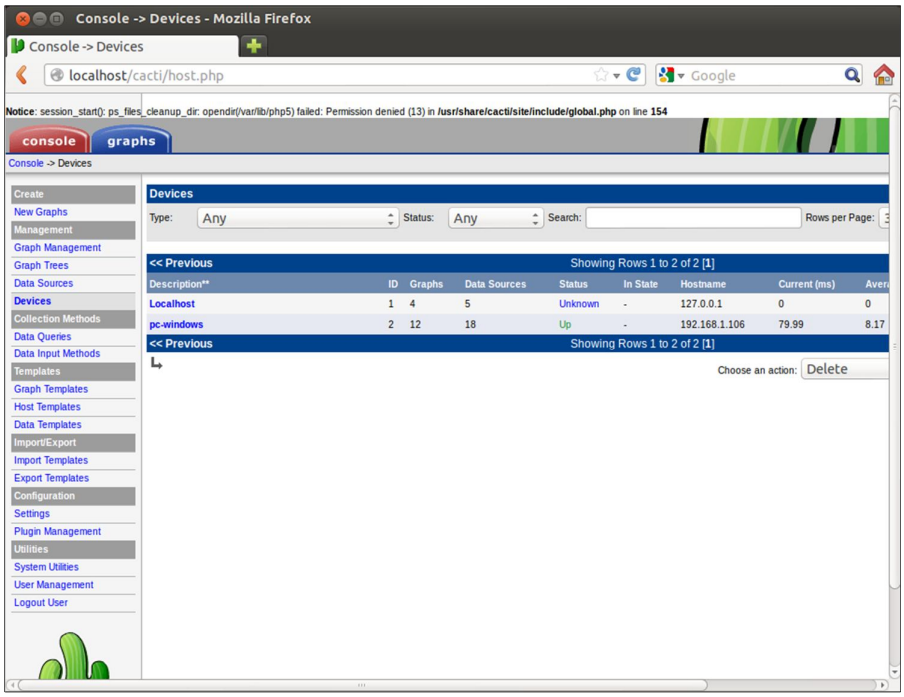

Figura 14 - Gerenciamento de dispositivos no Cacti
Fonte: The Cacti Group, Inc.

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 iy 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 15.

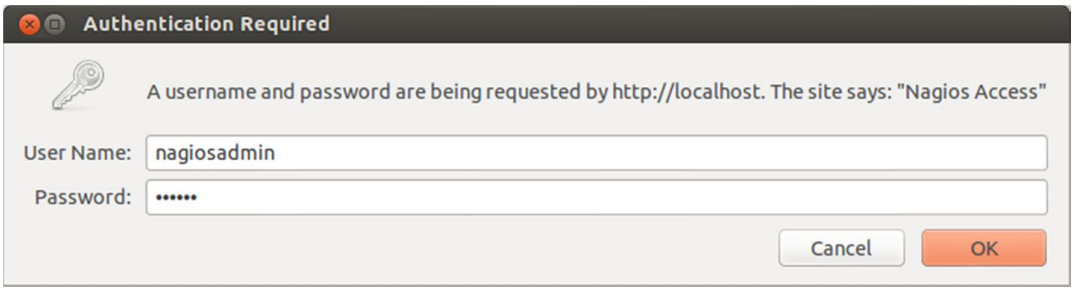

Figura 15 - Login do Nagios
Fonte: Nagios Enterprises, LLC.

Na figura 16, é possível visualizar a tela inicial do Nagios onde pode-se acessar suas abas principais de recursos.

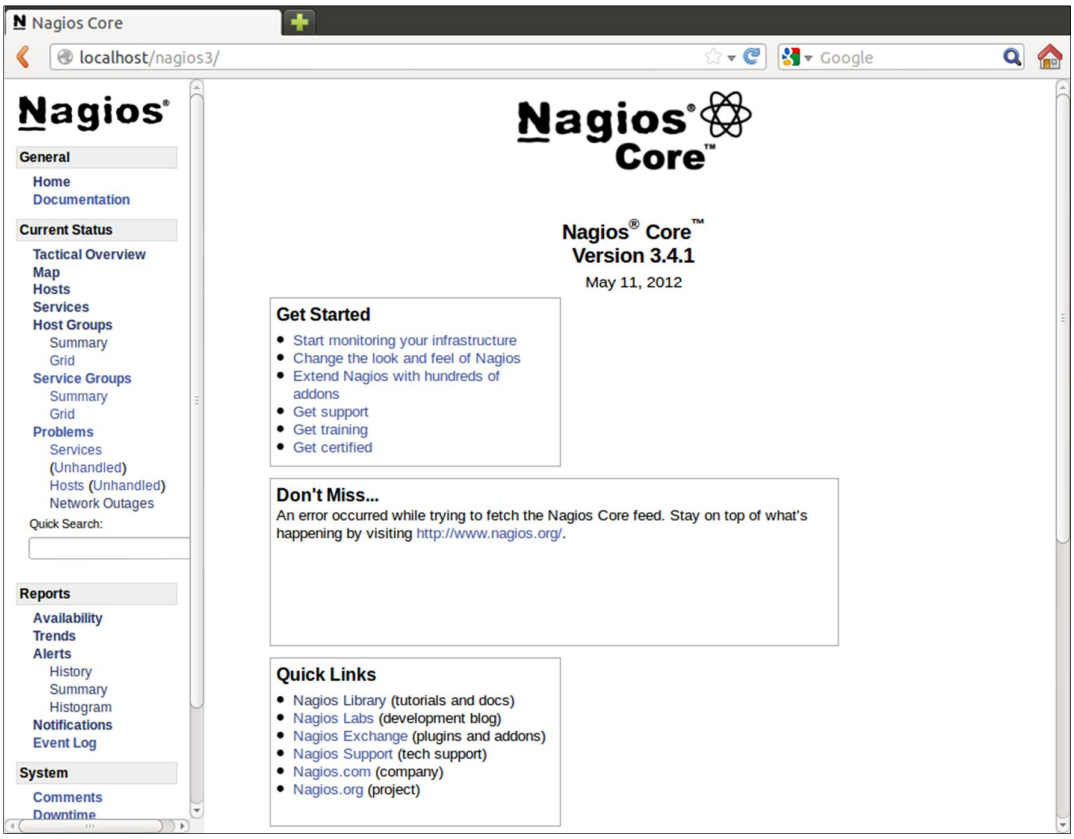

Figura 16 - Tela inicial do Nagios
Fonte: Nagios Enterprises, LLC.

Observando a Figura 17, é mostrado um resumo geral do gerenciamento e seu desempenho 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 como detecção de flap, notificações, manipuladores de eventos, verificações ativas.

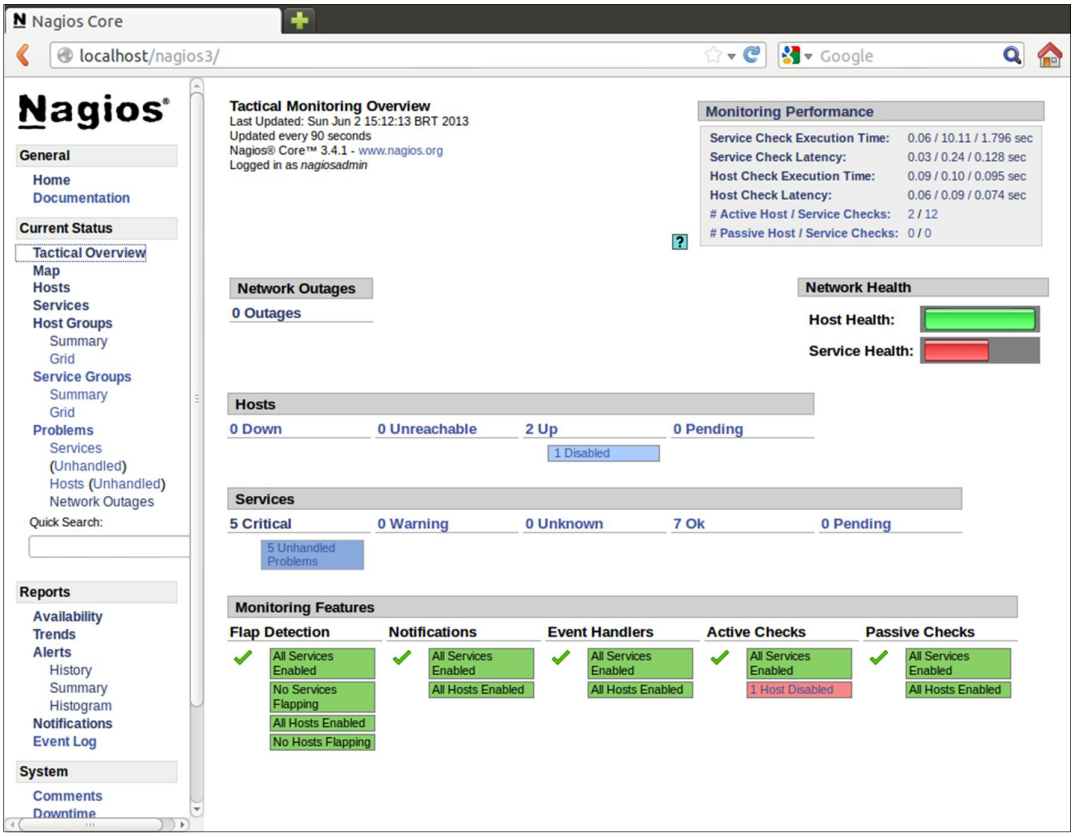

Figura 17 - Resumo do gerenciamento do Nagios
Fonte: Nagios Enterprises, LLC.

Na Figura 18, o Nagios mostra o detalhamento do host e dos serviços que ele utiliza, informando e notificando 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.

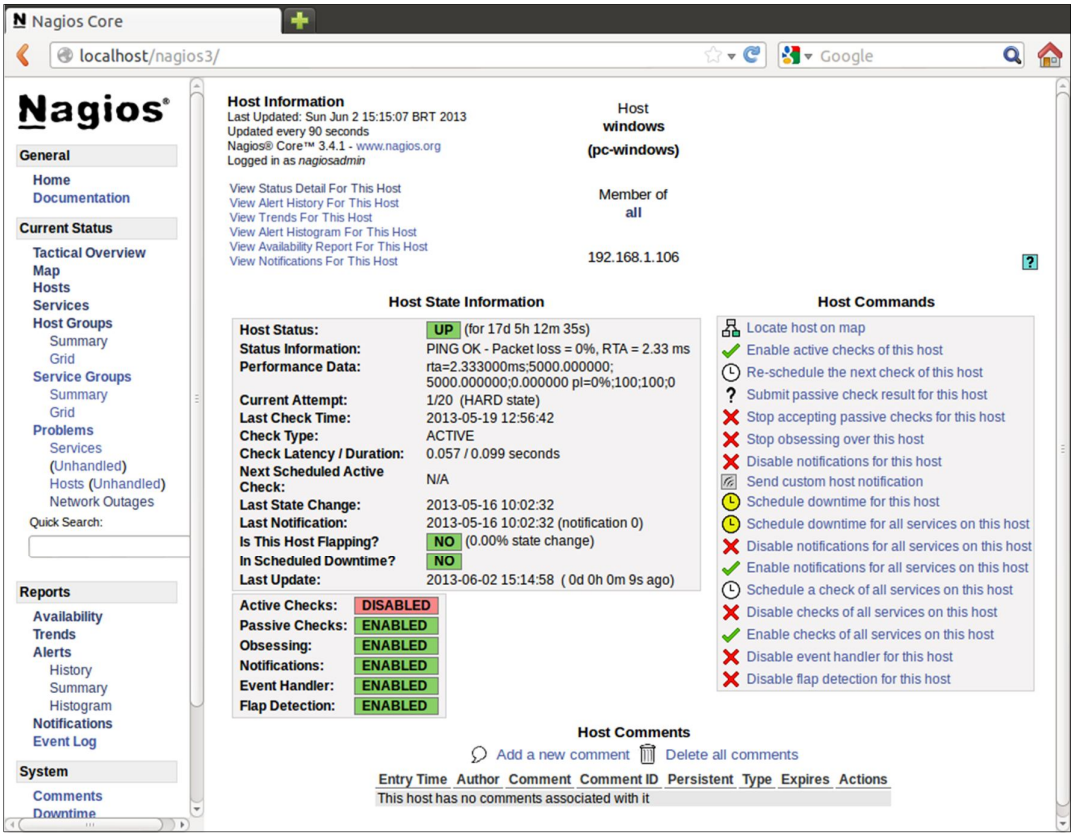

Figura 18 - Detalhamento do host no Nagios
Fonte: Nagios Enterprises, LLC.

Nesta parte do programa, observando a Figura 19, o Nagios permite o acesso a relatórios detalhados em relação ao gerenciamento 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.

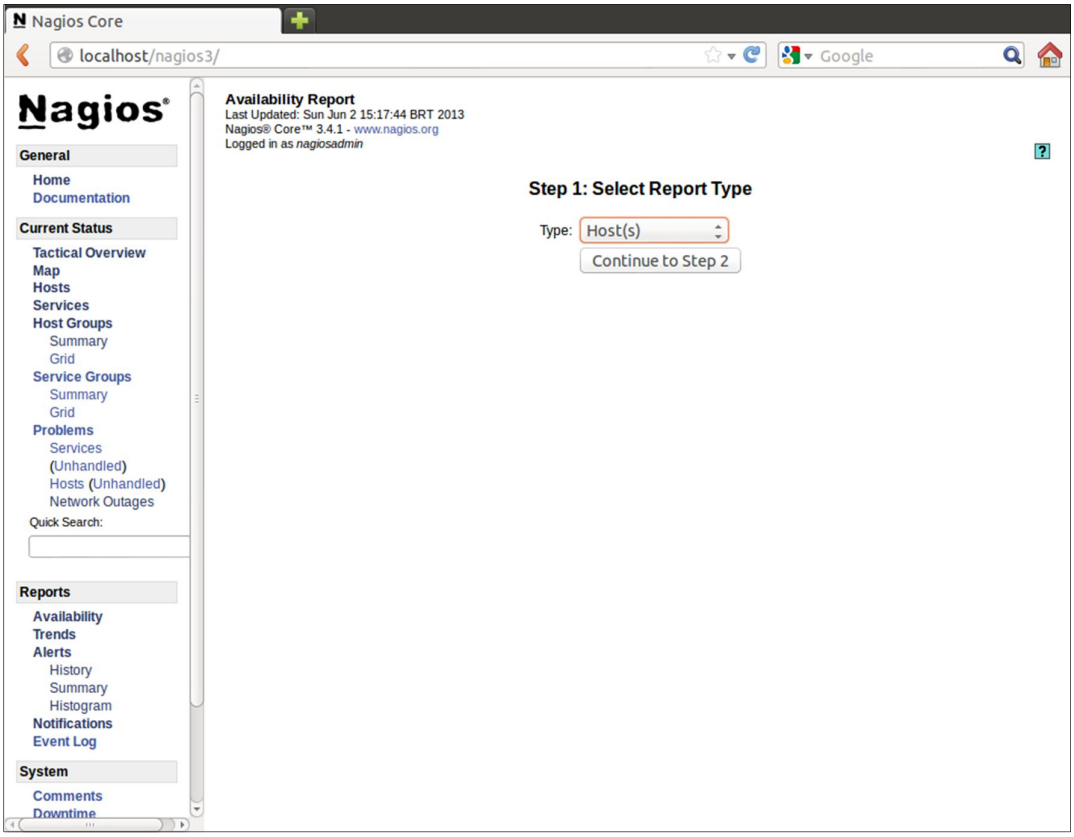

Figura 19 - Relatório Disponibilidade Nagios (passo 01)
Fonte: Nagios Enterprises, LLC.

No segundo passo, conforme Figura 20, são escolhidos os serviços a serem relatados.

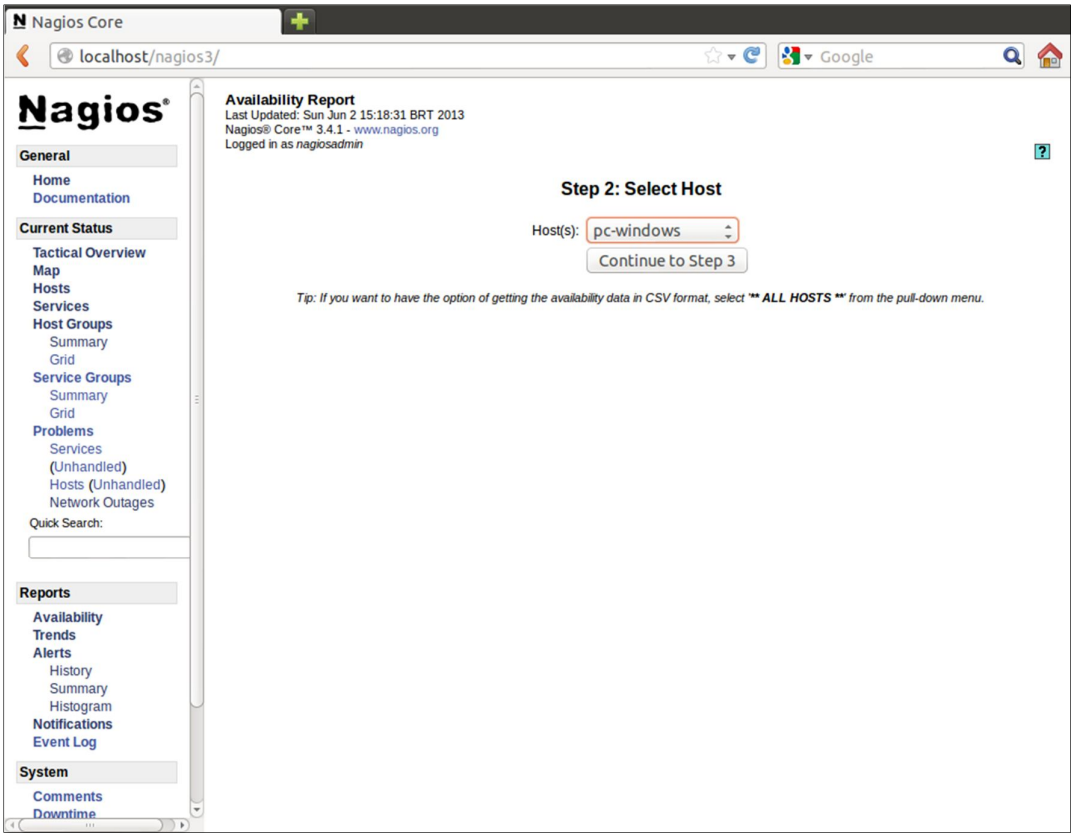

Figura 20 - Relatório Disponibilidade Nagios (passo 02)
Fonte: Nagios Enterprises, LLC.

Já no terceiro passo, conforme Figura 21, é escolhido o período de tempo em que os serviços foram gerenciados.

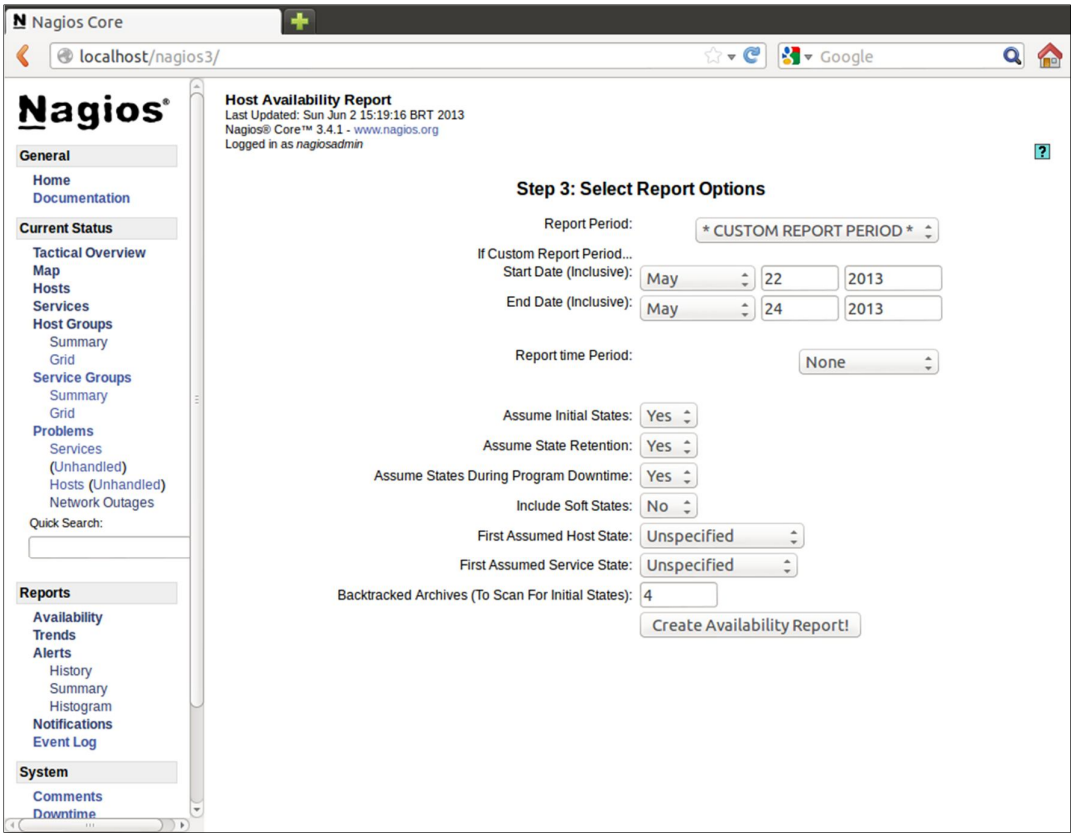

Figura 21 - Relatório Disponibilidade Nagios (passo 03)
Fonte: Nagios Enterprises, LLC.

E por fim é gerado o relátorio com todas as opções escolhidas detalhadas pelo gerenciamento, como podemos ver na Figura 22.

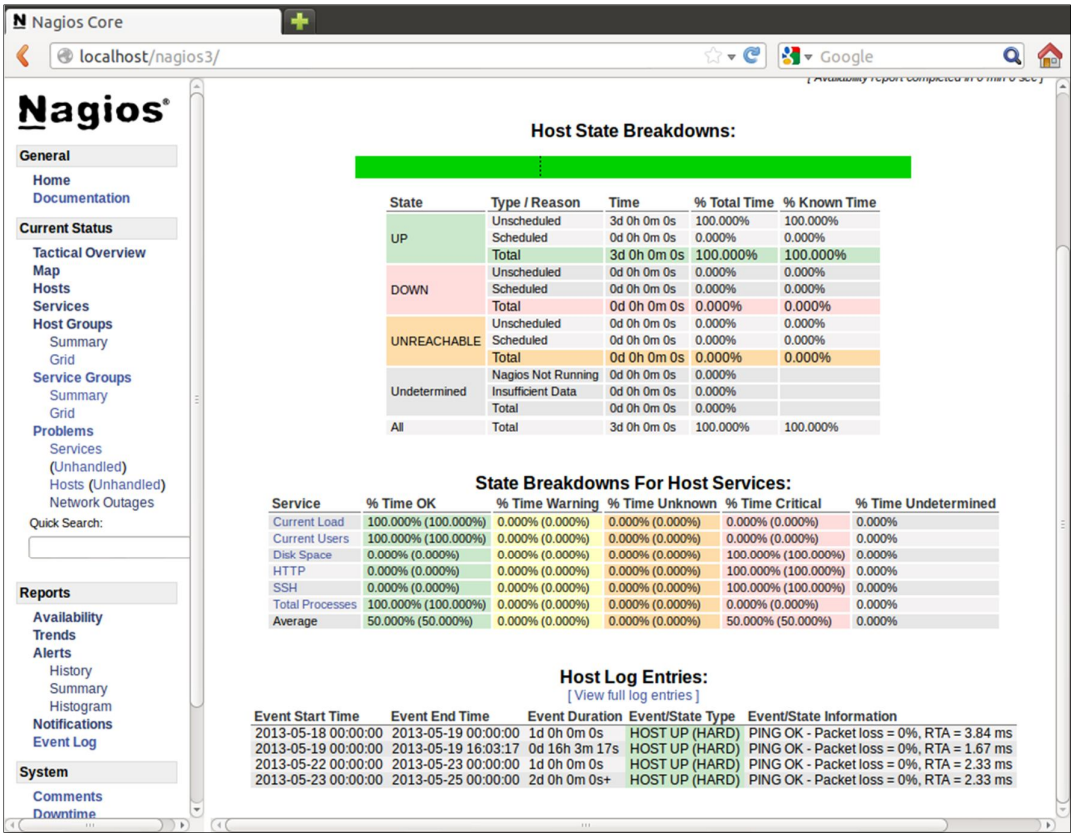

Figura 22 - Relatório Disponibilidade Nagios (resultados)
Fonte: Nagios Enterprises, LLC.

Outro recurso do Nagios, conforme mostra Figura 23, os Event Logs (log de eventos), 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.

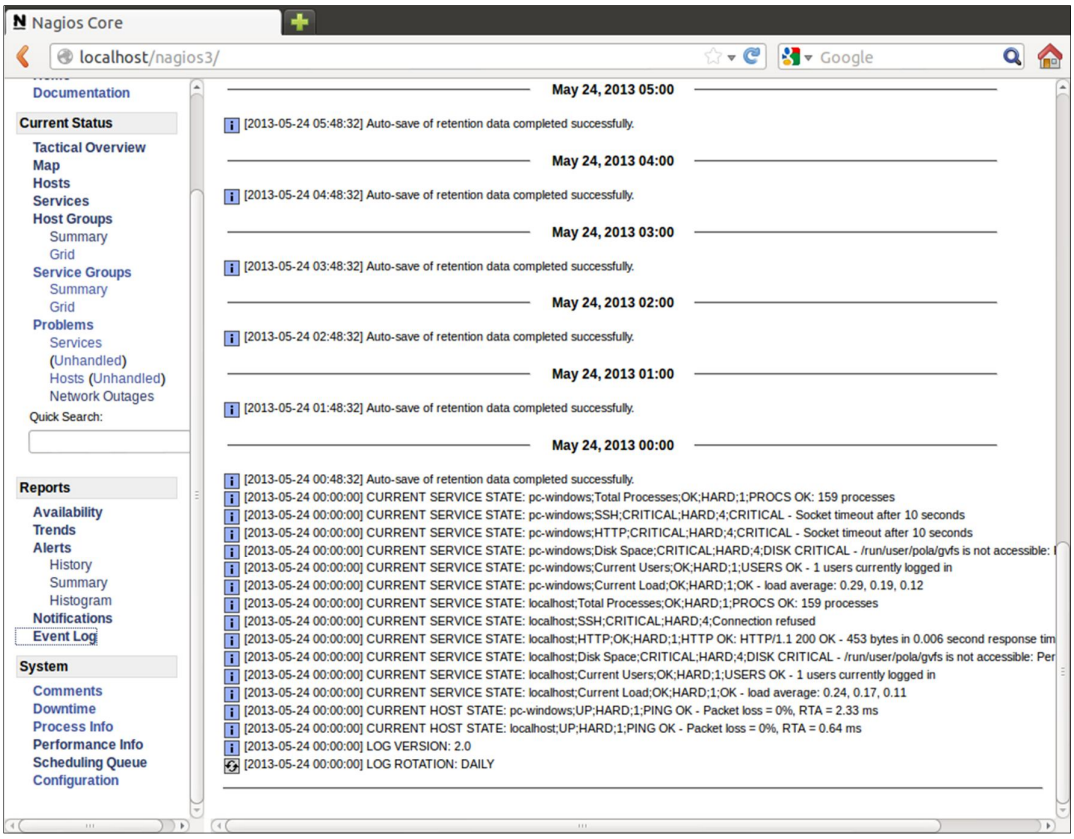

Figura 23 – Event Log do Nagios
Fonte: Nagios Enterprises, LLC.

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. Como vemos na figura 24.

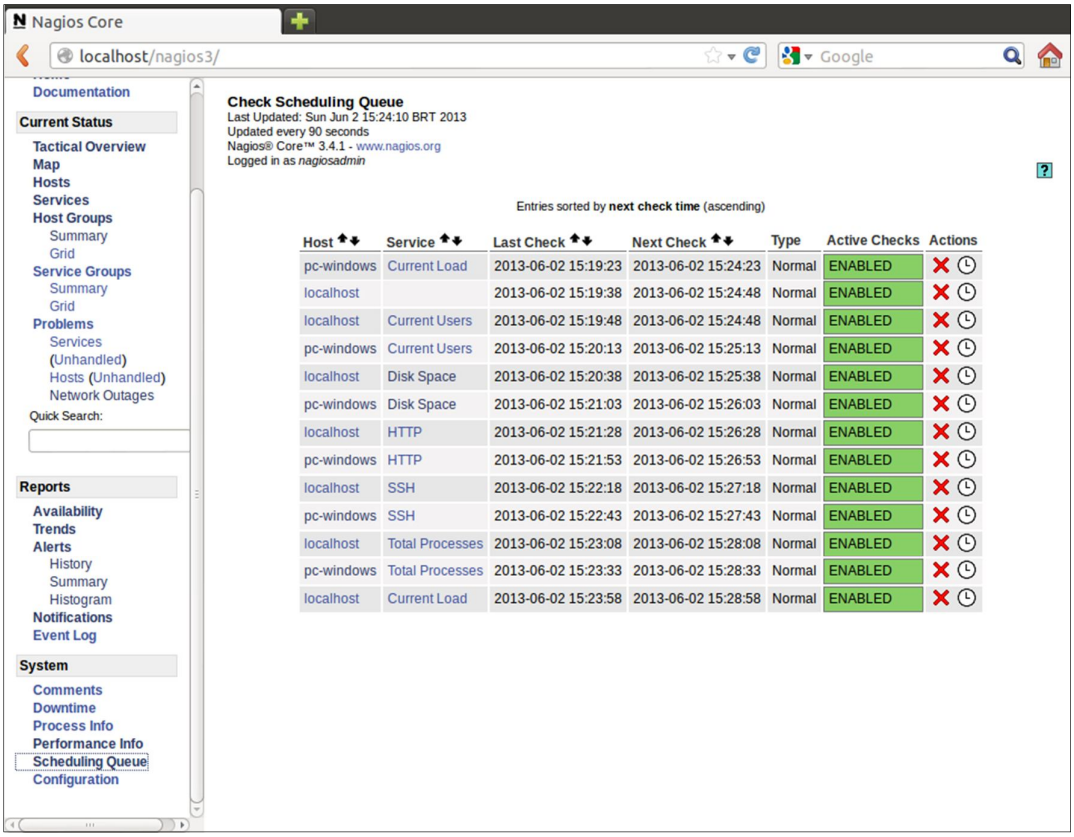

Figura 24 - Agendamento de serviços do Nagios
Fonte: Nagios Enterprises, LLC.

7. CONSIDERAÇÕES FINAIS

Depois de aplicadas as observações de gerenciamento e feitas às 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.

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

Para o gerenciamento 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 gerenciar os mais variados serviços e plataformas.

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 possui muita eficiência. Por isso ele não sobrecarrega o servidor nem os dispositivos de rede; permite que outras aplicações possam compartilhar os dados SNMP; o teste de dispositivos é feito de forma rápida; gera relatórios identificando imprecisões existentes e possui dados de configuração unificados.

As ferramentas estudadas apresentam muitas semelhanças entre si e, em geral, fornecem soluções para a maioria das necessidades que o gerenciamento de redes exige.

Por fim, os dois softwares analisados neste trabalho mostraram possuir muitas opções para um gerenciamento 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 resultado satisfatório e funcional, sem contar o fato de serem totalmente gratuitos. Como podemos ver na figura abaixo:

Nagios X Cacti		
Quesitos	**Cacti**	**Nagios**
SLA Reports	Não	Através de Plugin
Auto Discovery	Através de Plugin	Através de Plugin
Agente	Não	Sim
SNMP	Sim	Sim
Syslog	Não	Sim
Permite Scripts Externos	Sim	Sim
Plugins	Sim	Sim
Linguagem que foi escrito	PHP	Perl
Alertas	Sim	Sim
Font-end Web	Controle Completo	Controle Parcial
Monitoramento Distribuido	Sim	Sim
Inventário	Através de Plugin	Através de Plugin
Armazenamento de Dados	RRDTool, MySQL	MySQL, MSSQL
Licenciamento	GPL	GPL
Geração de Gráficos	Sim	Sim
Mapas	Através de Plugin	Sim
Eventos	Através de Plugin	Sim

Figura 2- Comparativo Nagios x Cacti.
Fonte – O autor.

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, il.

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 2013.

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 2013.

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

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

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

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

DYE, MCDONALD, Mark A, Network **Fundamentals: CCNA exploration companion guide**. ed. Cisco Press. Indianapolis, 2009.

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

GOETEN, Luciano Waltrick; STRINGARI, Sergio; UNIVERSIDADE REGIONAL DE BLUMENAU, Centro de Ciências Exatas e Naturais. **Protótipo de um software agente SNMP para rede Windows**. , 2001. 73p, il. Orientador: Sérgio Stringari.

HARNEDY, Sean. Total SNMP: **Exploring the Simple Netowrk 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; PERICAS, Francisco Adell; UNIVERSIDADE REGIONAL DE BLUMENAU. **Prototipo de um agente SNMP para uma rede local utilizando a plataforma JDMK**. , 2000. x, 88p, il. Orientador: Francisco Adell Pericas.

RIGNEY, Steve. **Planejamento e gerenciamento de redes**. 1ª edição. Editora Campus, 1996.

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

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.

SCHULZ, Murilo Alexandre. **Protótipo de *software* de gerência de desempenho de um access point de rede sem fio utilizando o protocolo SNMP.** 2004. Trabalho de Conclusão de Curso (Engenharia de Telecomunicações) – Centro de Ciências Tecnológicas, Universidade Regional de Blumenau, Blumenau.

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

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

TEIXEIRA, Ramos. **Redes de Computadores, serviços, administração e segurança**. 1ª edição. Editora Makron Books, 1999.