UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

YURI ANDREOLI

ANÁLISE COMPARATIVA ENTRE FERRAMENTAS PARA GERENCIAMENTO E MONITORAMENTO DE REDES

TRABALHO DE CONCLUSÃO DE CURSO

PATO BRANCO
2016

YURI ANDREOLI

ANÁLISE COMPARATIVA ENTRE FERRAMENTAS PARA GERENCIAMENTO E MONITORAMENTO DE REDES

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Conclusão de Curso 2, do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, da Universidade Tecnológica Federal do Paraná, Câmpus Pato Branco, como requisito parcial para obtenção do título de Tecnólogo.

Orientador: Prof. Alisson Andrey Puska

PATO BRANCO
2016

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Pato Branco
Departamento Acadêmico de Informática
Curso de Tecnologia em Análise e Desenvolvimento de Sistemas

TERMO DE APROVAÇÃO

TRABALHO DE CONCLUSÃO DE CURSO

ANÁLISE COMPARATIVA ENTRE FERRAMENTAS PARA GERENCIAMENTO E MONITORAMENTO DE REDES

por

YURI ANDREOLI

Este trabalho de conclusão de curso foi apresentado no dia 22 de novembro de 2016, como requisito parcial para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, pela Universidade Tecnológica Federal do Paraná. O acadêmico foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho APROVADO.

Banca examinadora:

Prof. Me. Alisson Andrey Puska
Orientador

Prof. Dr. Eden Ricardo Dosciatti

Prof. Dr. Fábio Favarim

Prof. Dr. Edilson Pontarolo
Coordenador do Curso de Tecnologia em Desenvolvimento de Sistemas

Profª. Me. Soelaine Rodrigues Ascari
Responsável pela Atividade de Trabalho de Análise e Conclusão de Curso

A folha de aprovação assinada encontra-se na Coordenação do Curso.

RESUMO

ANDREOLI, Yuri. Análise Comparativa Entre Ferramentas para Gerenciamento e Monitoramento de Redes. 2016. 108 f. Monografia (Trabalho de Conclusão de Curso) - Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, Universidade Tecnológica Federal do Paraná, Câmpus Pato Branco. Pato Branco, 2016.

Pode-se afirmar que redes de computadores são cada vez mais importantes no cotidiano das pessoas e que impactam diretamente no processo de globalização, nos negócios, e, certamente, no desenvolvimento das sociedades. A fim de que as redes de computadores consigam desempenhar suas funções e possam atingir seus objetivos, é fundamental que se faça o acompanhamento do desempenho das mesmas. Para isso, são utilizadas ferramentas de gerenciamento e monitoramento de redes, as quais foram abordadas durante este trabalho, em uma análise feita em um ambiente de rede de um provedor de Internet. Essa análise teve o objetivo de definir qual das ferramentas é a mais adequada para o cenário escolhido, abordando o gerenciamento de falhas, o gerenciamento de desempenho e o monitoramento em três ferramentas gratuitas: Cacti, OMD – *Open Monitoring Distribution* – e Zabbix, utilizando como critérios de avaliação, a instalação das ferramentas, a configuração de *hosts* e serviços e a capacidade de gerar informações claras e objetivas.

Palavras-chave: Redes de computadores. Gerenciamento de falhas. Gerenciamento de desempenho. Monitoramento.

ABSTRACT

ANDREOLI, Yuri. 2016. Comparative Analysis Between Tools for Management and Monitoring of Networks. 108 p. Monograph (Final Paper). Technologist Degree on System Analysis and Development. Federal University of Technology – Pato Branco, 2016.

It is possible to state that computer networks are becoming widely important to people's daily lives and that they have a direct impact on many areas, such as the globalization process, business and the development of societies. In order to allow computer networks to perform its functions and to achieve its goals, it is of fundamental importance that we obtain means of monitoring its performance; therefore, there is a necessity for tools for management and monitoring of these networks. The purpose of this is study is to analyze these tools inside of a network environment in an Internet provider. This analysis aims at defining which tool is more adequate to the chosen scenario, taking into account the examination of the monitoring process, and the management of failure and performance. For that purpose, three free tools – Cacti, OMD (Open Monitoring Distribution) and Zabbix – are going to be evaluate according to the following criteria: the setup of the tool, the settings of host and service, and the capability of generating clear and objective data.

Keywords: Computer networks. Fault management. Performance management. Monitoring.

LISTA DE FIGURAS

LISTA DE QUADROS E TABELAS

LISTA DE SIGLAS, ABREVIATURAS E ACRÔNIMOS

ARP	*Address Resolurion Protocol*
ASN.1	*Abstract Syntax Notation 1*
CMIP	*Common Management Information Protocol*
CPU	*Central Processing Unit*
EGP	*Exterior Gateway Protocol*
FAQ	*Frequently Asked Questions*
GB	*Gigabyte*
GHz	Giga-hertz
GNU	*General Public License*
GUI	*Graphical User Interface*
IAB	*Internet Architeture Board*
IANA	*Internet Assigned Numbers Authority*
ICMP	*Internet Control Message Protocol*
IETF	*Internet Engineering Task Force*
IP	*Internet Protocol*
IPMI	*Intelligent Platform Management Interface*
ISO	*International Organization for Standardization*
LAMP	Linux, Apache, MySQL, PHP
LSM	*Linux Security Modules*
MAC	*Media Access Control*
Mbps	Megabit por segundo
MD5	*Message Digest 5*
MIB	*Management Information Base*
Ms	Milissegundos
MySQL	*My Structured Query Language*
OMD	*Open Monitoring Distribution*
OSI	*Open System Interconection*
PDU	*Protocol Data Unit*
PHP	*Hypertext Preprocessor*
RAM	*Random Access Memory*
RFC	*Request for Comments*
RRD	*Round Robin Database*
SELinux	*Security Enhanced Linux*
SGMP	*Simple Gateway Management Protocol*
SHA1	*Secure Hash Algotithm 1*
SMI	*Structure and Identification of Management Information for TCP/IP-Based Internets*
SMS	*Short Message Service*
SMTP	*Simple Mail Transfer Protocol*
SNMP	*Simple Network Management Protocol*
SSL	*Secure Socket Layer*
TB	*Terabyte*
TCP/IP	*Transmission Control Protocol/Internet Protocol*
UDP	*User Datagram Protocol*
VPN	*Virtual Private Network*
XML	*eXtensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO

Nesta Seção são apresentadas as considerações iniciais, os objetivos, a justificativa e a estrutura do trabalho.

1.1 CONSIDERAÇÕES INICIAIS

A necessidade de comunicação entre os seres humanos é de suma importância para o desenvolvimento das sociedades, ainda mais em um mundo globalizado como o atual. Um fator importantíssimo e facilitador para que essa comunicação ocorra são as redes de computadores. São elas que possuem o objetivo de interligar cidades, países e continentes através de meios digitais.

Com toda essa responsabilidade atribuída às redes de computadores, pode-se afirmar que problemas envolvendo-as interferem diretamente no ciclo de trabalho e evolução das sociedades. Por isso, algumas características são fundamentais para o bom funcionamento do sistema, segundo Forouzan (2006, pg. 34), entre elas: a entrega, em que o sistema deve entregar os dados ao destino correto, e a confiabilidade, ou seja, o sistema deve fazer o melhor possível para assegurar a entrega dos dados sem modificações.

O administrador de redes é quem possui o papel de assegurar essas características às redes de computadores. Para isso, utiliza ferramentas que o auxiliam neste processo, como, por exemplo, ferramentas de monitoramento e gerenciamento de redes. Uma apresentação funcional no processo de gerenciamento foi apresentada pela ISO (*International Organization for Standardization*) dividindo o gerenciamento de redes em cinco áreas (Falhas, Configuração, Contabilização, Desempenho e Segurança).

Este trabalho aborda, de forma mais enfática, o gerenciamento de falhas e o gerenciamento de desempenho, visando encontrar a melhor solução para o cenário em que as ferramentas foram implantadas, examinando os seguintes critérios: A instalação das ferramentas; a configuração de *hosts* e serviços; e a capacidade de gerar informações claras e objetivas, sendo que cada critério conta com suas respectivas métricas.

1.2 OBJETIVOS

Nesta Seção são apresentados os objetivos do trabalho, sendo agrupados em objetivo geral e objetivos específicos.

1.2.1 Objetivo Geral

- Realizar um ensaio comparativo entre três softwares livres – OMD (*Open Monitoring Distribution*), Cacti e Zabbix – utilizados no gerenciamento de redes de computadores, a fim de determinar qual tem o melhor comportamento no ambiente escolhido para o ensaio.

1.2.2 Objetivos Específicos

- Implantar as soluções para o ensaio em um ambiente de rede funcional de um provedor de Internet;
- Auxiliar o gerenciamento e monitoramento da rede através da solução escolhida ao final do ensaio;
- Indicar a ferramenta mais adequada aos administradores de redes, com base nos resultados provenientes desta análise.

1.3 JUSTIFICATIVA

Para Forouzan (2006, pg. 37), a definição de uma rede é descrita como: “(...) conjunto de dispositivos (denominados nós) conectados por *links* de comunicação. Um nó pode ser um computador, uma impressora ou qualquer outro dispositivo capaz de enviar e/ou receber dados gerados em outros nós da rede”. Moura (2005) resume que as ferramentas de gerenciamento e monitoramento são utilizadas para o “acompanhamento dos eventos de uma rede, a fim de diagnosticar problemas e determinar quando e quais procedimentos de contingência devem ser aplicados,

bem como obter estatísticas para administração e otimização de desempenho". Stallings (1999) define softwares de gerência de redes como um sistema que pode ser composto por outras ferramentas, tendo como função auxiliar no serviço de gerência de redes, coletando informações dos dispositivos e oferecendo facilidades através de comandos.

Sobre o gerenciamento de falhas, Specialski (1999, p. 3) define que "uma falha é uma condição anormal cuja recuperação exige ação de gerenciamento. Uma falha normalmente é causada por operações incorretas ou um número excessivo de erros". O monitoramento da rede é importante para que, segundo Specialski (1999, p. 3), após o aparecimento de uma falha, seja possível: rapidamente determinar o componente em que a falha ocorreu, isolar a rede do nó com falha, reconfigurar ou modificar a rede para minimizar o impacto da operação sem o nó problemático e reparar ou trocar o componente com falha.

Com as citações referidas, pode-se concluir que o gerenciamento de falhas é fundamental para que a rede gerenciada consiga manter-se disponível e com qualidade. Há uma enorme quantidade de ferramentas capazes de fazer esse gerenciamento, cada uma com suas qualidades e limitações. Devido a esse fator, definir qual ferramenta é melhor para cada cenário acaba se tornando uma tarefa difícil. O provedor de Internet onde foram implantadas as ferramentas não possuía um estudo detalhado de qual se adequaria melhor ao cenário de rede que possui, por isso este trabalho fez-se necessário. O ensaio abordou apenas softwares livres, visto que a rede abordada não possui muitos nós e o provedor é de médio porte, o que tornaria inviável o investimento em ferramentas pagas para gerenciar a rede.

O referido trabalho possui grande relevância em sua relação com os conhecimentos adquiridos durante o curso, visto que para o desenvolvimento do mesmo, foram necessários conhecimentos sobre todos os aspectos que os softwares de gerenciamento empregam. A análise de softwares de gerenciamento não depende apenas do conhecimento em redes de computadores, mas também leva em conta, por exemplo, a necessidade de utilização de lógicas de programação para a criação de *scripts* personalizados para monitorar determinados serviços, a utilização de métricas de qualidade de software dentro dos critérios avaliativos pré-estabelecidos, a utilização e gerenciamento de banco de dados, entre outros conhecimentos que foram envolvidos ao longo do trabalho.

1.4 ESTRUTURA DO TRABALHO

O presente trabalho está estruturado em seis capítulos. O primeiro deles apresenta as considerações iniciais, os objetivos e a justificativa. O segundo capítulo, por sua vez, traz o referencial teórico, contendo uma introdução sobre o gerenciamento de redes, as áreas funcionais do gerenciamento do modelo OSI (*Open System Interconection*) e os padrões de gerenciamento com seus respectivos protocolos. O terceiro capítulo trata do cenário em que foram implantadas as ferramentas e dos materiais utilizados. O quarto capítulo apresenta os critérios avaliativos e métricas utilizados para avaliar as ferramentas. O quinto capítulo expõe os detalhes das avaliações, os resultados de cada uma das métricas e também o resultado final. O sexto e último capítulo traz a conclusão do trabalho e os trabalhos futuros.

2 GERENCIAMENTO DE REDES

Independentemente do tamanho de uma rede de computadores, é fundamental que a mesma seja gerenciada. Somente dessa forma pode-se alcançar uma eficiência no uso dos recursos e dos serviços oferecidos. O profissional responsável pela gerência de redes é o administrador de redes e, para que esse profissional possa desempenhar uma gerência de qualidade, é fundamental que o mesmo seja auxiliado por ferramentas específicas para essa função. Algumas definições sobre gerenciamento de redes são expostas a seguir, bem como os elementos que fazem parte do gerenciamento.

Para Forouzan e Mosharraf (2013, p. 693) o gerenciamento de redes pode ser definido como “(...) a tarefa de testar, monitorar, configurar e resolver problemas dos componentes de rede com o objetivo de atender um conjunto de requisitos definidos por uma organização”. Stallings (1999) define que o objetivo da gerência de redes é garantir um nível de qualidade de serviço através do monitoramento e controle dos elementos que a compõem, tanto físicos, como lógicos.

Forouzan e Mosharraf (2013, p. 693) definem ainda que “(...) um sistema de gerenciamento de rede utiliza hardware, softwares e seres humanos”. Dessa forma, pode-se definir como hardware os elementos que foram gerenciados e a estação de gerência; como softwares, as ferramentas que foram estudadas neste trabalho; e como seres humanos, as pessoas envolvidas no trabalho de implantação, análise e coleta dos dados providos pelas ferramentas que foram analisadas. Lopes, Sauvé e Nicoletti (2002, p. 17) mostram que um sistema de gerenciamento possui quatro componentes básicos, que são:

- Elementos gerenciados: Equipamento que possui um software (agente) que permite o monitoramento do mesmo através de uma estação de gerência.
- Estação de gerência: Possui um software chamado gerente que comunica-se diretamente com os agentes dos elementos gerenciados. Ainda segundo Lopes, Sauvé e Nicoletti (2002, p. 17), “a estação de gerência oferece uma interface através da qual usuários autorizados podem gerenciar a rede”.

- Protocolo de gerência: Protocolo de comunicação entre agente e gerente para fins de monitoramento (leitura) e controle (escrita). Um exemplo é o protocolo SNMP (*Simple Network Management Protocol*).
- Informações de gerência: Dados que trafegam entre agente e gerente através do protocolo de gerência.

A Figura 1 apresenta os elementos que fazem parte de um sistema de gerência de redes:

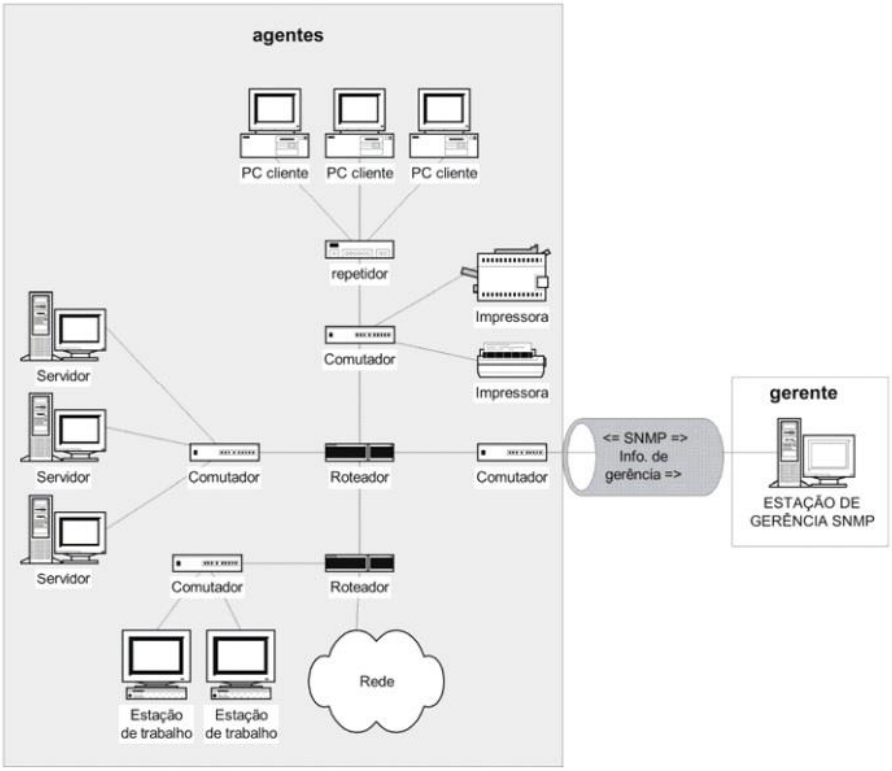


Figura 1 - Elementos da arquitetura de um sistema de gerência
Fonte: Lopes, Sauvé e Nicoletti (2003, p. 18)

2.1 ÁREAS FUNCIONAIS DE GERENCIAMENTO DO MODELO OSI

A ISO divide o gerenciamento de redes em cinco áreas distintas como parte de sua especificação de Gerenciamento de Sistemas para o modelo de rede OSI. São elas: gerenciamento de falhas, de configuração, de contabilização, de desempenho e de segurança. Um bom sistema de gerenciamento de redes aborda todas as áreas definidas, já que as mesmas impactam diretamente na rede gerenciada. Cabe ressaltar que este trabalho se focou na capacidade das ferramentas em realizar o gerenciamento de falhas e de desempenho da rede estudada.

2.1.1 Gerenciamento de falhas

Eler (2015, p. 4) define o gerenciamento de falhas com a “função de monitorar os estados dos recursos verificando em qual ponto da rede e quando uma falha ou um erro pode ocorrer”. Specialski (1999, p. 3) ressalta que “falhas não são o mesmo que erros”. Isso porquê, tanto para Eler (2015, p. 4) quanto para Specialski (1999, p. 3), uma falha é uma condição anormal persistente e que causa a total interrupção da rede, como, por exemplo, o rompimento de um cabo; uma falha exige uma ação de gerenciamento imediata. Um erro é uma condição anormal ocasional que pode ser corrigida ou compensada antes que se transforme em uma falha, tendo como exemplo, erro de bits durante uma transmissão.

O gerenciamento de falhas tem um alto grau de impacto na rede, visto que, diferentemente das falhas, erros normalmente não são previsíveis. Forouzan (2007, p. 875) ainda subdivide o gerenciamento de falhas em “gerenciamento de falhas reativo” e “gerenciamento de falhas proativo”.

O “gerenciamento de falhas proativo” tenta prevenir a ocorrência de falhas. Para isso, o administrador da rede busca informações que possam ajudá-lo a impedir que eventuais erros se transformem em falhas, causando a interrupção total da rede e afetando diretamente os que a utilizam.

O “gerenciamento de falhas reativo” consiste em tomar atitudes somente após a ocorrência da falha. Forouzan (2007, p. 875) define que o “gerenciamento de falhas reativo” possui quatro atribuições: detecção, isolamento, correção e documentação. As mesmas são descritas a seguir:

- Detectar o exato local onde a falha ocorreu – Só após determinar o local exato em que a falha ocorreu é possível adotar as outras ações.
- Isolar a falha do restante da rede – Esse procedimento é importante para diminuir a quantidade de usuários afetados.
- Corrigir a falha – Após detectar e isolar a falha, é necessário fazer a etapa principal do gerenciamento de falhas, que é a correção da mesma.
- Documentar a falha – A documentação é importante para que se possa fazer um estudo mais detalhado do que está acontecendo, como por exemplo, se é necessária ou não a troca de um equipamento que já falhou outras vezes. Além disso, falhas iguais podem reaparecer com o tempo, com uma documentação completa (contendo onde aconteceu a

falha, a possível causa e como a mesma foi corrigida), a ação de reparo tende a ser mais rápida.

2.1.2 Gerenciamento de configuração

Para Eler (2015, p. 4), o gerenciamento de configuração “permite manter atualizadas as informações de hardware e software de uma rede, incluindo as informações de configurações de todos os equipamentos”. Isso é necessário, visto que redes de computadores estão em constantes mudanças, tanto de hardwares quanto de softwares. Pode-se afirmar, portanto, que com todas as mudanças que acontecem nas redes de computadores, é de suma importância que haja uma gerência de configurações, para que, por exemplo, em caso de troca de um equipamento que possua *backup* de suas configurações, o tempo de substituição do mesmo seja reduzido, gerando menor transtorno em uma rede ativa.

2.1.3 Gerenciamento de contabilização

Specialski (1999, p. 3) define que o “administrador da rede deve estar habilitado para controlar o uso dos recursos por usuário ou grupo de usuários”. Isso é necessário para que não haja monopolização da rede por parte de um usuário ou grupo de usuários que abuse dos seus privilégios de acesso e para evitar um uso ineficiente da rede por parte dos mesmos. A contabilização também tem a função de registrar o consumo da rede para que haja cobrança pelo mesmo.

2.1.4 Gerenciamento de desempenho

Segundo Forouzan (2007, p. 876), o gerenciamento de desempenho tem o objetivo de monitorar e controlar a rede para assegurar que a mesma funcione da forma mais eficiente possível. O gerenciamento de desempenho se correlaciona com o gerenciamento de falhas, de modo que uma gerência de desempenho mal executada pode ocasionar erros na rede e, como descrito, erros na rede podem

ocasionar falhas. O gerenciamento de desempenho é importante para que, por exemplo, o administrador da rede possa tomar atitudes a fim de ampliar determinado segmento da rede que está sofrendo com algum tipo de gargalo, como a falta de banda disponível ou uma latência excessiva, impedindo erros tais quais a perda de pacotes ou redução da largura de banda efetiva dos usuários.

2.1.5 Gerenciamento de segurança

Forouzan (2007, p. 876) cita que o objetivo do gerenciamento de segurança é controlar o acesso à rede baseado em uma política de segurança pré-determinada. Specialski (1999, p. 4) ainda ressalta a necessidade da existência de uma política de segurança “robusta e efetiva e que o sistema de gerenciamento de segurança seja, ele próprio, seguro”.

Neste trabalho abordou-se de forma mais intensa o gerenciamento de falhas e o gerenciamento de desempenho, que foram as maiores exigências para o cenário estudado.

2.2 PADRÕES DE GERENCIAMENTO E SEUS PROTOCOLOS

As redes se tornam cada dia mais complexas e com uma diversidade maior de equipamentos e fabricantes. Para que haja o gerenciamento das mesmas, é necessário também que todos os equipamentos consigam comunicar-se de forma padrão com a estação de gerência. Com isso, surgiram padrões de gerenciamento e protocolos. Os dois protocolos que mais se destacam no mercado são o SNMP (*Simple Network Management Protocol*) e o CMIP (*Common Management Information Protocol*).

O protocolo CMIP trabalha na camada de aplicação do modelo OSI e é orientado à conexão. O CMIP também utiliza o conceito de gerente e agente, porém, consome mais recursos da rede do que o SNMP. Por ser mais complexo, algumas redes não o suportam (BARRIVIERA, 2010).

Neste trabalho levou-se em conta de forma mais completa o protocolo SNMP. Este protocolo é amplamente utilizado no monitoramento de dispositivos,

sendo um protocolo padrão pertencente ao conjunto de protocolos da Internet definido pela IETF (*Internet Engineering Task Force*). Todos os equipamentos monitorados pela análise deste trabalho possuem suporte nativo ao protocolo SNMP.

2.2.1 Arquitetura de gerência SNMP

A arquitetura de gerenciamento SNMP, segundo Case et al. (1990, p. 2), consiste em: MIB (*Management Information Base*), que tem a função de descrever quais são os objetos contidos na mesma; SMI (*Structure and Identification of Management Information for TCP/IP-based Internets*), que descreve como os objetos gerenciados contidos na MIB são definidos; e o próprio SNMP, que é o protocolo a ser utilizado para gerenciar estes objetos. A Figura 2 representa a arquitetura de gerenciamento SNMP.

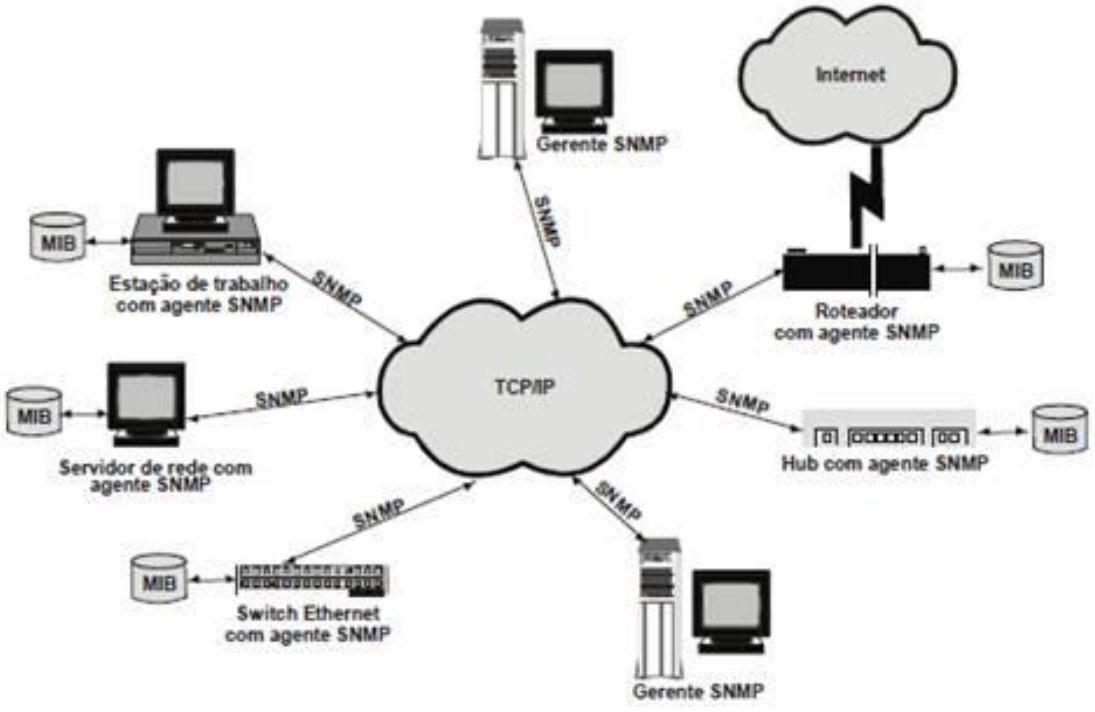


Figura 2 - Arquitetura de gerência SNMP
Fonte: Poleto (2012, p. 2)

2.2.1.1 Management Information Base – MIB

Um importante componente utilizado no gerenciamento de redes é a *Management Information Base* (MIB), que atualmente está em sua segunda versão (MIB2), definida na RFC (*Request for Comments*) 1213. Segundo Forouzan (2007, p. 886), a MIB nada mais é do que uma coleção de todos os objetos que o gerente pode manipular no agente. A MIB2 é uma versão atualizada da MIB1 e fornece informações gerais sobre o equipamento gerenciado. Os objetos contidos na MIB2 são categorizados em dez diferentes grupos. O Quadro 1 apresenta esses grupos e suas descrições.

Objeto	**Identificador do Objeto**	**Descrição**
System	1.3.6.1.2.1.1	Define uma lista de objetos que pertencem ao sistema operacional (*uptime*, contato, nome do sistema, etc)
Interfaces	1.3.6.1.2.1.2	Mantém o status de cada interface da entidade que está sendo gerenciada (*up* ou *down*, octetos enviados e recebidos, etc)
AT	1.3.6.1.2.1.3	Informações a respeito da tabela ARP (*Address Resolution Protocol*)
IP (*Internet Protocol*)	1.3.6.1.2.1.4	Mantém várias informações a respeito de IP, incluindo rotas
ICMP (*Internet Control Message Protocol*)	1.3.6.1.2.1.5	Informações a respeito do protocolo ICMP (erros, pacotes enviados e recebidos, etc)
TCP (*Transmission Control Protocol*)	1.3.6.1.2.1.6	Informações gerais sobre o protocolo TCP (tabela de conexão, *time-out*s, etc)
UDP (*User Datagram Protocol*)	1.3.6.1.2.1.7	Informações gerais sobre o protocolo UDP (número de portas, número de pacotes enviados e recebidos, etc)
EGP (*Exterior Gateway Protocol*)	1.3.6.1.2.1.8	Informações gerais sobre o protocolo EGP
Transmission	1.3.6.1.2.1.10	Reservado para MIBs específicas
SNMP	1.3.6.1.2.1.11	Informações gerais sobre o protocolo SNMP

Quadro 1 - Grupos da MIB2 e suas descrições
Fonte: Adaptado de Douglas e Schmidt (2005, p. 36)

Todos esses grupos pertencem ao MIB2 e são utilizados pelo protocolo SNMP, porém, ainda há outros dois tipos de MIB: a MIB experimental, utilizada para objetos que estão em fase de desenvolvimento e testes; e a MIB privada, que contém objetos específicos do elemento gerenciado.

2.2.1.2 Structure and Identification of Management Information for TCP/IP-based Internets – SMI

Dentro da arquitetura SNMP, o SMI tem a responsabilidade de definir os objetos contidos na MIB. O SMI possui duas versões, sendo que a primeira foi apresentada na RFC1155 e é denominada SMIv1. Já a segunda, apresentada na RFC2578, é denominada SMIv2 e apresenta melhorias para a segunda versão do protocolo SNMP, além de possuir novos tipos de dados, como o *Counter64*.

Segundo Forouzan (2007, p. 881), o SMI tem três funções: nomear os objetos, definir o tipo de dados que são armazenados em determinado objeto e exibir o método de codificação para os dados a serem transmitidos.

Nome: O SMI define que cada objeto gerenciado possua um nome único. Para nomear os objetos, o SMI usa um identificador de objeto que é baseado em uma estrutura hierárquica de árvore. A Figura 3 apresenta o exemplo de uma árvore de identificação de objetos.

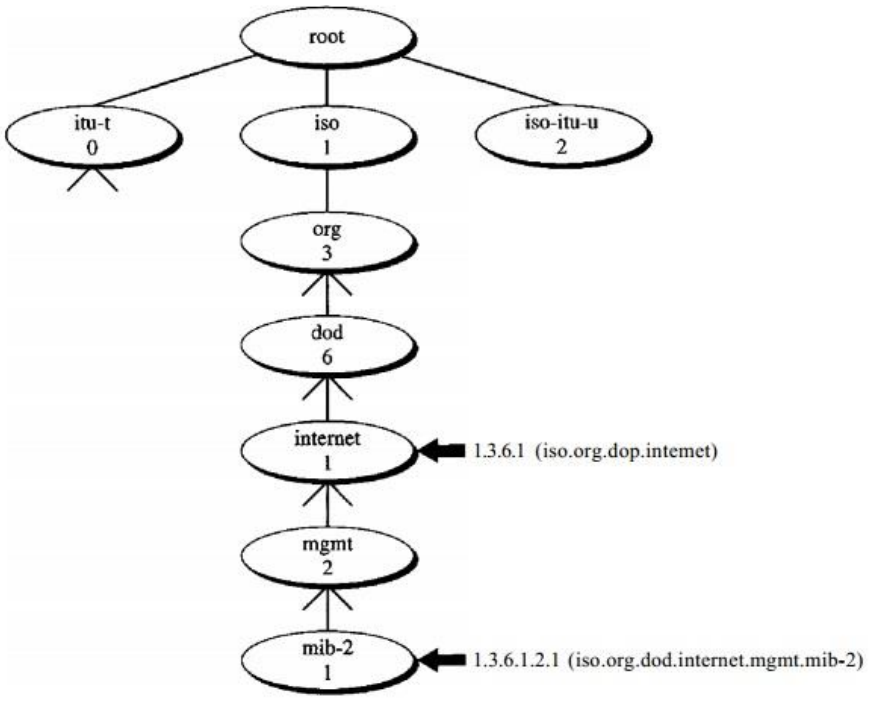


Figura 3 - Árvore de nomes de identificador de objetos
Fonte: Forouzan (2007, p. 882)

A árvore possui em seu topo o nó chamado *root*, todos os galhos abaixo são filhos do nó *root*. Forouzan (2007, p. 882) define que cada objeto é definido por uma sequência de inteiros separados por ponto, sendo que cada inteiro representa um

galho da árvore. Além da sequência de inteiros, também há uma sequência de nomes separada por pontos. O protocolo SNMP utiliza-se da sequência de inteiros, enquanto a sequência de nomes é usada apenas pelas pessoas. Por padrão, o protocolo SNMP utiliza os objetos contidos no galho mib-2, ou seja, o identificador começa com: 1.3.6.1.2.1 (iso.org.dod.internet.mgmt.mib-2).

Douglas e Schmidt (2005, p. 25) citam que além dos identificadores contidos no galho *mgmt*, há identificadores que podem ser cedidos a empresas privadas. A regulamentação desses identificadores para empresas privadas é feita pela IANA (*Internet Assigned Numbers Authority*). Por exemplo, os identificadores privados da Cisco são definidos com: 1.3.6.1.4.1.9 (iso.org.dod.internet.private.enterprises.cisco).

Tipo: O segundo atributo pelo qual o SMI é responsável é o tipo de dados dos objetos. O SMI possui dois tipos de dados: simples (dados atômicos que não possuem nenhuma relação entre si, como inteiros, caracteres, *strings*) e estruturados (dados constituídos por uma estrutura de dados simples que possuem relação entre seus valores, como os *structs* e *arrays*).

Alguns dos dados do tipo simples são extraídos da ASN.1 (*Abstract Syntax Notation One*); outros, adicionados pela própria SMI. Já os dados estruturados podem ser *"sequência"* e *"sequência de"*. O primeiro é uma combinação de dados simples não necessariamente do mesmo tipo, como um *struct* da linguagem de programação C, enquanto o segundo é uma combinação de dados simples, porém de um mesmo tipo, ou uma combinação de tipos de *"sequência"*, que utiliza o mesmo conceito dos *arrays* (FOROUZAN, 2007, p. 883).

O Quadro 2 apresenta os tipos de dados do SMIv2.

Tipo	Tamanho	Descrição
INTEGER	4 *bytes*	Um inteiro com o valor entre -2^{31} e 2^{31-1}
Integer32	4 *bytes*	O mesmo que o INTEGER
Unsigned32	4 *bytes*	Valor entre 0 e 2^{32-1}
OCTET STRING	Variável	Sequência de até 65,535 *bytes*
OBJECT IDENTIFIER	Variável	Um identificador de objeto
IPAddress	4 *bytes*	Um endereço IP composto de quatro inteiros
Counter32	4 *bytes*	Inteiro que pode ser incrementado de 0 até 2^{32} e quando atingir o máximo, volta ao valor de 0
Counter64	8 *bytes*	Contador de 64 *bits*
Gauge32	4 *bytes*	Inteiro que pode ser incrementado de 0 até 2^{32} e quando atingir o máximo não volta ao valor de 0, ao menos que seja zerado
TimeTicks	4 *bytes*	Contador que registra tempo em centésimo de segundos
BITS		Sequência de *bits*
Opaque	Variável	Sequência não interpretada

Quadro 2 - Tipos de dados do SMIv2
Fonte: Adaptado de Forouzan (2007, p. 883)

Método de codificação: O SMI utiliza o BER (*Basic Encoding Rules*) para codificar os dados a serem transmitidos na rede. O BER é um conjunto de regras de codificação ASN.1 “que define a forma através da qual um programa escrito nessa linguagem é compilado para ser traduzido para a linguagem de máquina do dispositivo da rede” (BERNAL, 2014, p. 2).

2.2.1.3 *Simple Network Management Protocol* – SNMP

Specialski (1990, p. 13) descreve que o protocolo SNMP começou a ser desenvolvido na década de 80. Ele surgiu com o intuito de aprimorar e suceder o SGMP (*Simple Gateway Management Protocol*) que objetivava apenas a gerência de *gateways.* Com o SNMP é possível gerenciar desde uma impressora até um roteador de borda.

O SNMP foi desenvolvido pelo IETF e teve sua primeira publicação no RFC 1067, de agosto de 1988. Em maio de 1990, a RFC 1157 publicou a versão SNMPv1. A segunda versão do protocolo, o SNMPv2, foi publicado na RFC 1901, de janeiro de 1996 e incluiu melhorias, como a possibilidade de uma implementação de gerenciamento distribuído, novos objetos MIBs e alteração nos nomes e formatos de operações já existentes. A última versão do SNMP, o SNMPv3, foi publicado na RFC 2571, em abril de 1999, incluindo ao protocolo algumas questões de segurança e padronização da implementação dos agentes e gerentes, utilizando o SMI e a MIB do SNMPv2 (LOBÃO, 2011).

Os objetivos principais do protocolo SNMP são: minimizar o número e a complexidade das funções de gerenciamento, de modo que todas as funções do protocolo sejam bem documentadas e de fácil acesso; ser suficientemente flexível para permitir futuras expansões da rede, já que a maioria dos dispositivos de rede possuem suporte nativo ao protocolo e não ficariam sem gerenciamento; por fim, ser independente da arquitetura e mecanismo dos dispositivos a serem gerenciados.

Poleto (2012, p. 2) cita que “o SNMP é um protocolo relativamente simples e robusto, porém suficientemente poderoso para resolver os difíceis problemas apresentados quando se deseja gerenciar redes heterogêneas”. O SNMP pode ser considerado um protocolo simples, pois os elementos de redes a serem gerenciados não necessitam um alto nível de processamento. Isto ocorre, pois, as tarefas

complexas de processamento não ficam a cargo do recurso gerenciado, mas sim da estação de gerência.

O protocolo SNMP trabalha na camada de aplicação do modelo TCP/IP (*Transmission Control Protocol/Internet Protocol*) e faz uso do protocolo UDP na camada de transporte para o envio das mensagens entre o gerente e os agentes. Case et al (1990, p. 15) especifica que o SNMP utiliza a porta 161 da estação de gerência para enviar solicitações aos agentes e receber as respostas, e a porta 162 do gerente para receber *traps* (informações importantes enviadas de forma automática dos agentes para o gerente).

A principal característica do protocolo UDP é não implementar a confiabilidade, ou seja, o emissor (gerente) não sabe se o receptor (agente) recebeu determinado pacote. Isso é uma vantagem considerável pois gera um menor consumo de recursos de rede, já que há menos mensagens de controle. Para que essa característica não afete o gerenciamento, Douglas e Schmidt (2005, p. 19) afirmam que a simples implementação de um sistema de *timeout* resolve este problema, e esse sistema de *timeout* deve ser implementado da seguinte maneira: o administrador da rede deve configurar em sua estação de gerência o tempo de resposta que o gerente deverá aguardar por uma resposta do agente, caso esse tempo de resposta expire, o gerente deve reenviar a solicitação. O número de reenvios que o gerente faz também deve ser configurado previamente pelo administrador.

Segundo Poleto (2012, p. 2), o agente é quem responde às requisições do gerente. O agente interage diretamente com a MIB do equipamento gerenciado e busca as informações requeridas pelo gerente. Além disso, o agente também é responsável por disparar *traps* ao gerente.

Poleto (2012, p. 2) ainda define os gerentes SNMP como softwares que são executados nas estações de gerenciamento, responsáveis por enviar os *requests* aos agentes e receber as respostas. Os gerentes ainda podem acessar as informações dos dispositivos gerenciados e modificá-las.

A Figura 4 mostra a relação entre gerentes e agentes SNMP.

Figura 4 - Relação entre agente e gerente SNMP
Fonte: Poleto (2012, p. 2). Adaptado pelo autor.

O SNMP possui várias operações que podem ser executadas. O PDU (*Protocol Data Unit*) é o formato de mensagem que gerentes e agentes usam para trocar informações. De acordo com Poleto (2012, p. 2), o SNMPv1 possui as seguintes operações:

- *Get-request*: Mensagem do gerente ao agente solicitando “o valor de uma ou mais variáveis contidas na MIB do elemento gerenciado”.
- *Get-next-request*: Mensagem do gerente ao agente que permite uma sequência de comandos para recuperar um grupo de valores a partir de uma MIB.
- *Set-request*: Mensagem do gerente ao agente que permite atribuir valor a uma variável da MIB.
- *Get-response*: Mensagem do agente ao gerente que responde os comandos *get*, *get-next* e *set*.
- *Trap*: Mensagem do agente ao gerente sem um *request*, que envia “informações de alarmes ou eventos significativos”.

A Figura 5 representa a troca de PDUs entre gerente e agente.

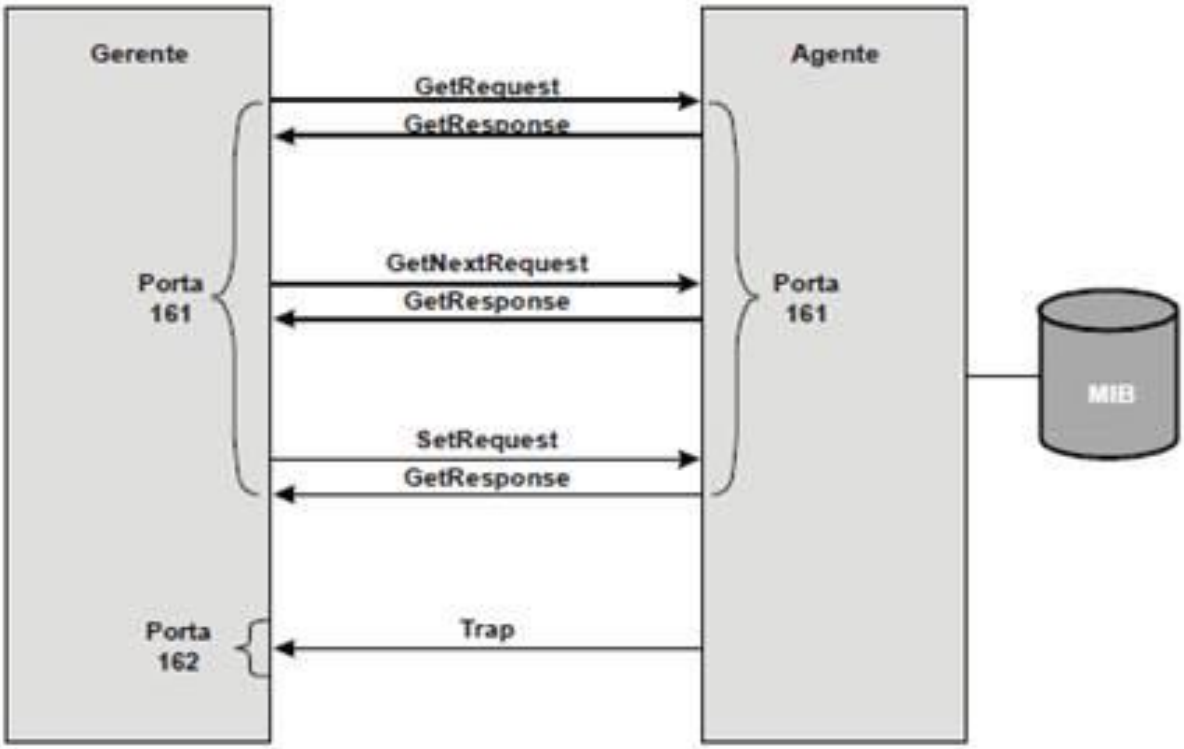


Figura 5 - Troca de PDUs entre gerente e agente SNMP
Fonte: Poleto (2012, p. 2)

Poleto (2012, p. 2) afirma que a segunda versão do SNMP (SNMPv2) possui algumas melhorias em relação ao SNMPv1, sendo as principais: implementação de melhorias de segurança; possibilidade de mais de um gerente e a troca de informações entre esses gerentes, permitindo a gerência distribuída; transferência de grandes blocos de informação; contadores de 64 bits; melhorias no tratamento de erros das variáveis da PDU; e, também, duas novas operações em relação ao SNMPv1.

As novas operações do SNMPv2 são:

- *Get-bulk-request*: Acesso a grandes blocos de informações na MIB.
- *Inform-request*: Mensagem enviada diretamente de um gerente a outro.

O SNMPv2 também alterou o nome de duas operações do SNMPv1, o *getresponse* passou a ser apenas *response*, e o *trap* passou a ser *snmpV2-trap*.

Douglas e Schmidt (2005, p. 73) reiteram que o maior problema com o protocolo SNMP sempre foi a segurança, já que, nas duas primeiras versões, a autenticação entre a estação de gerência e os agentes era feita apenas por uma senha, chamada de *community*; essa senha é transferida entre o gerente e o agente

de forma textual pura, sem nenhum tipo de criptografia. O principal objetivo de se criar uma nova versão do SNMP foi justamente melhorar a segurança, sendo que o SNMPv3 possui um método de autenticação que utiliza os algoritmos de criptografia MD5 (*Message Digest 5*) e SHA1 (*Secure Hash Algorithm 1*), substituindo o texto puro da *community* (DOUGLAS; SCHMIDT; 2005, p. 80).

Além da criptografia, o SNMPv3 implementa algumas outras regras de segurança, porém, não implementa nenhuma nova operação ao protocolo, suportando apenas as operações já conhecidas do SNMPv1 e do SNMPv2. O SNMPv3 ainda modificou alguns conceitos e terminologias, a principal delas foi o abandono da ideia de agente e gerente, passando a chamá-los de entidades SNMP. (DOUGLAS, SCHMIDT; 2005, p. 74).

Este trabalho utilizou o SNMPv2 como protocolo padrão de gerenciamento, visto que as estações de rádio monitoradas não possuem suporte ao SNMPv3. Além disso, todos os equipamentos possuem uma *community* definida previamente pelo administrador da rede.

3 CENÁRIO E MATERIAIS

Nesta Seção são destacados os materiais e métodos utilizados para a realização do trabalho, além de uma descrição do cenário em que foi realizada a análise.

3.1 CENÁRIO

O programa Cidade Digital, de São Lourenço do Oeste - SC, foi criado e é mantido com recursos próprios do município, e está em funcionamento desde 2011. O programa possui o objetivo de levar o acesso à Internet banda larga a qualquer morador da cidade, sem a necessidade de pagamento de mensalidade, além de pontos de acesso gratuito em praças, restaurantes, bibliotecas, entre outros locais públicos. O Cidade Digital também contempla as escolas municipais de São Lourenço do Oeste.

O provedor de Internet em que as foram implantadas as ferramentas possui a responsabilidade de manter o programa Cidade Digital em pleno funcionamento, através do gerenciamento da rede do programa, manutenções preventivas, manutenções corretivas e instalações de novos pontos de acesso aos interessados. Não é de responsabilidade do provedor a instalação de novas estações de rádio para ampliar a cobertura do sinal. Este trabalho tem o intuito de auxiliar o provedor na medida em que, identificados gargalos na rede, o mesmo possa fazer as correções necessárias para melhorar o desempenho da rede e também para determinar qual das ferramentas abordadas neste estudo é a mais adequada para o cenário.

A rede do programa Cidade Digital é toda interligada via fibra óptica e centralizada no *Data Center* da própria Prefeitura Municipal. A rede possui um *uplink* de 100Mbps (*Megabits* por segundo), sendo que cada usuário que adere ao programa possui uma limitação de banda de 1Mbps. Atualmente, a rede conta com treze estações de rádio 2.4GHz (Giga-hertz) Wavion WBS-2400, distribuídas estrategicamente para que haja cobertura de sinal *wi-fi* em toda a área urbana da cidade. Também é utilizado um *switch* Cisco Catalyst WS-C2960-24LC-S, um *WebFilter* Barracuda 610 para fazer a filtragem de conteúdo, um MikroTik

Rouberboard 1200 e um servidor *Radius* com dois processadores Xeon E5620 @ 2.40Ghz e 4GB (*Gigabyte*) de memória RAM (*Random Access Memory*).

Como estação de gerência foi utilizado uma máquina *desktop* com um CPU (*Central Processing Unit*) Athlon II @ 2.7Ghz, 2GB de memória RAM, disco rígido SB600, com o sistema operacional CentOS Linux 7.2.1511.

A rede gerenciada foi acessada pela estação de gerência através de uma VPN (*Virtual Private Network*), “uma rede privada construída em uma infraestrutura de rede pública, como a Internet” (CISCO, 2016). Com o uso de VPN, a estação de gerência não precisou estar fisicamente conectada à rede citada no cenário. Para este trabalho não houve autorização por parte do provedor, para realizar o monitoramento do servidor *Radius*, do *WebFilter* e do MikroTik *Rouberboard*, portanto, foram analisadas apenas as estações de rádio e o *switch*. A Figura 6 apresenta o diagrama com todos os equipamentos que fazem parte do cenário.

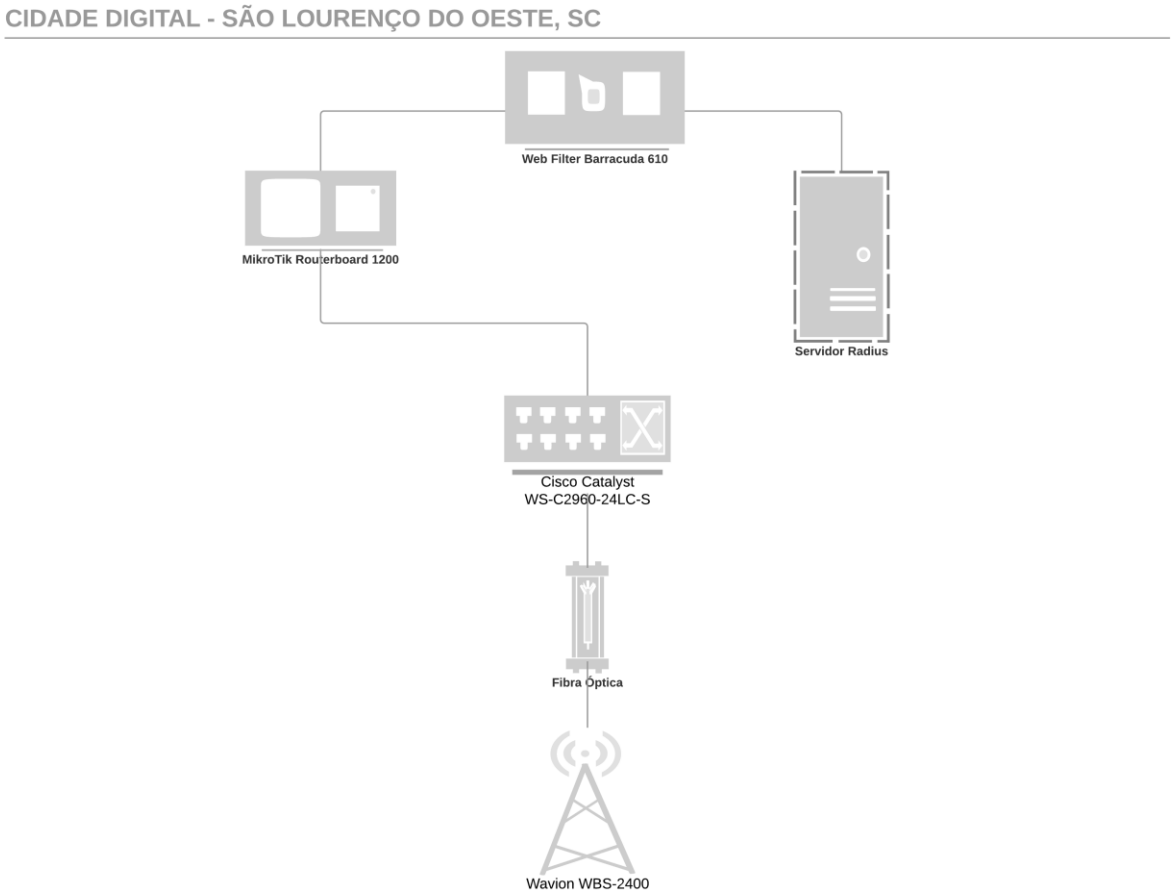


Figura 6 – Diagrama do cenário
Fonte: Autoria própria

3.2 MATERIAIS

Neste trabalho foram analisadas três ferramentas gratuitas para monitoramento e gerenciamento de rede. As três ferramentas e suas características são descritas abaixo.

3.2.1 OMD – *Open Monitoring Distribution*

O OMD é uma solução de monitoramento baseada no Nagios, um conhecido sistema *open-source* para monitoramento de redes. O OMD implementa um novo conceito de instalação, manutenção e atualização de um sistema de monitoramento construído sob o Nagios (OMD, 2016). O OMD tem a intenção de facilitar a instalação e configuração do Nagios, além de disponibilizar diversos *plug-ins* a fim de melhorar e facilitar o gerenciamento.

Uma das maiores dificuldades de se trabalhar com o Nagios é a instalação e configuração dos *hosts* e serviços, posto que todo esse trabalho é feito através de configurações em arquivos de texto, o OMD oculta este trabalho através de um *plug-in* já implementado, chamado Check_MK. O Check_MK é uma extensão para o Nagios que permite a criação de configurações através do uso da linguagem *Python*, permitindo maior flexibilidade e retirando alguns gargalos do Nagios. Adicionalmente, o Check_MK também implementa uma GUI (*Graphical User Interface*) que torna o Nagios mais “amigável” para o administrador da rede.

O OMD, além disso, implementa outros *plug-ins* que devem ser destacados: o NagVis, uma ferramenta que permite a criação de mapas e diagramas para a rede gerenciada, e o *pnp4nagios*, *plug-in* para a geração de gráficos da rede através da ferramenta RRD (*Round Robin Database*).

Atualmente, o OMD contém (OMD, 2016): Nagios (e diversos *plug-ins*), Icinga, Shinken, NagVis, pnp4nagios, rrdtool/rrdcached, Check_MK, MK Livestatus, Multisite, Dokuwiki, Thruk, Mod-Gearman, check_logfiles, check_oracle_heatlh, check_mysql_health, jmx4perl, check_webinject e check_multi.

Todas essas ferramentas e *plug-ins* possibilitam alguns recursos ao OMD, como:

- Múltiplas instâncias por *host*: em uma mesma estação de gerência pode haver mais de uma instância do Nagios. Essas instâncias são chamadas de *sites*;
- Separação de usuário por instância: cada *site* possui seus próprios usuários;
- Criação simples de novos *sites*: basta um comando para que seja criado um novo *site* na estação de gerência. Esse recurso pode ser útil para, por exemplo, ter um *site* no monitoramento e outro podendo ser usado para a realização de testes no ambiente;
- Diferentes versões do OMD ao mesmo tempo: possibilita ao administrador da rede instalar diferentes versões do OMD em uma mesma estação de gerência;
- Otimizações de recursos: através das ferramentas já citadas que o OMD implementa, a solução tenta reduzir principalmente a utilização de disco da estação de gerência.

O OMD pode ser instalado na maioria das distribuições Linux através de pacotes pré-compilados e atualmente está na versão 1.30, que foi a utilizada neste trabalho. Não foram encontrados requisitos mínimos exigidos pela ferramenta.

3.2.2 Cacti

O Cacti é uma ferramenta livre para monitoramento de redes, desenvolvida em PHP (*Hypertext Preprocessor*) que utiliza a plataforma RRDTool (*Round Robin Database Tool*) para criar gráficos e armazenar informações dos estados da rede em uma base de dados MySQL (*My Structured Query Language*). O Cacti é distribuído sob a licença GNU (*General Public License*), e também possui suporte a *plug-ins* externos (CACTI, 2016).

Alguns recursos do Cacti são:

- Gráficos: o Cacti possui um número ilimitado de gráficos, podendo ser configurados como gráficos de área, linha, barra, entre outros, além de

possibilitar ao administrador configurar cores, legendas e outros detalhes para facilitar a visualização do gráfico;

- Coleta de dados: o Cacti permite que o usuário faça a coleta de dados através de *scripts* externos, além do suporte ao protocolo SNMP.
- Gerenciamento de usuários: permite que o administrador crie usuários e aplique regras de permissões aos mesmos.

Há três operações básicas que o Cacti executa (CACTI, 2016):

- Obtenção dos dados: é feita através de um *poller*[1], que executa a aplicação em um intervalo de tempo constante para buscar os dados do SNMP ou dos *scripts* externos;
- Armazenamento dos dados: é realizado através do RRDTool, possibilitando a visualização de valores médios, mínimos e máximos, este recurso possibilita a redução da utilização de disco;
- Apresentação dos dados: um dos maiores recursos do RRDTool é a criação de gráficos dinâmicos, e o Cacti utiliza-se disso para apresentar os dados ao usuário.

O Cacti pode ser instalado tanto em sistemas Windows quanto em sistemas Linux, e, para as duas plataformas, exige alguns pré-requisitos, que são: RRDTool 1.0.49 ou superior, MySQL 4.1.x ou superior, PHP 4.3.6 ou superior e um web *server* como o Apache, por exemplo. Atualmente o Cacti está na versão 0.8.8h, sendo esta a versão implementada neste trabalho.

3.2.3 Zabbix

"O Zabbix é uma ferramenta de monitoramento de redes, servidores e serviços, pensada para monitorar a disponibilidade, experiência de usuário e qualidade de serviços" (4Linux, 2016). Vale também ressaltar que o Zabbix é uma ferramenta *open*-source capaz de, segundo Vladishev (2016, p. 16), suportar mais de cem mil dispositivos e alocar TB's (*terabytes*) de informações. As principais características do Zabbix são:

---

[1] *Script* ou programa que se comunica com os dispositivos para coletar as informações (CACTI, 2016).

- Suporte a maioria dos sistemas operacionais;
- Suporte a protocolos simples sem o uso de agentes (HTTP, POP3, IMAP);
- Possui suporte nativo ao protocolo SNMP;
- Gerenciamento através de uma interface *web*;
- Integração com banco de dados;
- Scripts personalizados.

Atualmente o Zabbix está na versão 3.2, sendo que a versão analisada foi a 3.0.5. Segundo a documentação oficial da ferramenta, para um monitoramento com menos de 100 *hosts* são necessários 128Mb de memória RAM e 256Mb livre em disco, mas esse valor pode variar dependendo da quantidade de *hosts* monitorados pela ferramenta.

4 CRITÉRIOS AVALIATIVOS E MÉTRICAS

Esta Seção é destinada à descrição dos critérios avaliativos utilizados para comparar as ferramentas analisadas, bem como as métricas de avaliação de cada critério.

4.1 CRITÉRIOS AVALIATIVOS

As três ferramentas a serem analisadas foram submetidas a alguns critérios de avaliação, sendo que cada uma das ferramentas foi analisada separadamente, ou seja, as ferramentas foram implantadas e analisadas uma a uma após uma nova instalação do sistema operacional na estação de gerência. Todas as ferramentas foram implantadas na estação de gerência supracitada e monitoraram os dispositivos do referido cenário durante um período de sete dias.

Foi considerada como a ferramenta mais adequada analisada dentro do cenário exposto, aquela que obteve um maior índice de primeiras colocações perante as métricas determinadas.

O Quadro 3 estabelece os critérios avaliativos aos quais as ferramentas foram submetidas.

Critério avaliativo	**Descrição**
Instalação das ferramentas	Foram analisadas e expostas as facilidades e dificuldades encontradas pelo autor durante a instalação das ferramentas
Configuração de *hosts* e serviços	Também foram analisadas e expostas a facilidades e dificuldades durante a configuração dos *hosts* e serviços monitorados
Geração de informações claras e objetivas	Este critério diz respeito a eficácia da ferramenta na geração de informações ao administrador da rede, como gráficos, diagramas de redes, entre outros

Quadro 3 - Critérios avaliativos
Fonte: Autoria própria

4.2 MÉTRICAS

Um processo de avaliação de software requer a definição clara de seus critérios, tal como as métricas que foram utilizadas para avaliar os softwares. A ISO/IEC/IEEE 24765 (2010, pg. 216) define “métrica” como sendo uma medida para avaliar se um sistema, componente ou processo possui determinado atributo.

Para compreender como foram feitas as análises das ferramentas é importante definirmos as classificações de métricas de avaliação, sendo elas: objetiva ou subjetiva e quantitativa ou qualitativa.

4.2.1 Métricas e suas classificações

Sato (2007, p. 42) define que métricas **objetivas** são aquelas cujo valor depende apenas do objeto em questão, não levando em consideração o ponto de vista de quem a está interpretando. Um exemplo de métrica objetiva é o número de pacotes necessários para a instalação de determinada ferramenta, já que o próprio terminal da máquina informa quantos pacotes serão instalados. Enquanto métricas **subjetivas** dependem do objeto em questão e do ponto de vista de quem a interpreta e avalia. Como exemplo de métrica subjetiva, pode-se citar o grau de dificuldade da instalação de uma determinada ferramenta em uma escala de 0 a 10, que apesar de possuir um valor numérico claro, depende da interpretação de quem a instalou.

Para Sato (2007, p. 43) “o valor de uma métrica **quantitativa** pertence a um intervalo de uma certa magnitude e geralmente é representado por um número”. Com base nessa definição, pode-se afirmar que as medidas quantitativas sempre podem ser comparadas entre si. Pode-se tomar como exemplo de métrica quantitativa deste trabalho o tempo de instalação das ferramentas. As métricas **qualitativas** são representadas não por números, mas sim por palavras, símbolos ou figuras. Um exemplo de métrica qualitativa usada durante o desenvolvimento do trabalho é a capacidade da ferramenta de enviar notificações ao administrador da rede.

4.2.2 Métricas utilizadas para a análise

A Seção a seguir tem o objetivo de descrever cada métrica, agrupando-as pelos critérios avaliativos e determinando as características e métodos utilizados para a obtenção das mesmas.

4.2.2.1 Métricas de instalação

Todas as métricas de instalação dizem respeito a instalação das ferramentas antes de qualquer configuração de *host* ou serviço, e foram avaliadas após o sistema operacional estar atualizado. As métricas de instalação podem ser vistas a seguir:

- Grau de dificuldade: métrica definida com o objetivo de classificar as ferramentas quanto ao grau de dificuldade do autor com a instalação das mesmas, considerada subjetiva/qualitativa. As ferramentas receberam avaliações de: **grau I**, quando foi necessário apenas o uso da documentação oficial e todos os passos resultaram na conclusão da instalação com êxito; **grau II**, quando os passos citados pela documentação oficial resultaram na conclusão da instalação, porém aconteceram erros durante a instalação e foi necessária uma pesquisa fora da documentação da ferramenta; e de **grau III,** em casos em que a instalação da ferramenta dependeu de pesquisas além da documentação oficial, e exigiu do autor maior conhecimento técnico, em termos de mudanças no *firewall* do sistema operacional, por exemplo. Quanto menor o grau de dificuldade, melhor a posição da ferramenta, informação esta que pode ser visualizada no Quadro 4, da Seção 5.1.1 do Capítulo 5;
- Necessidade de pacotes adicionais: essa métrica objetiva/quantitativa tem o objetivo de expor quantos pacotes adicionais são necessários para a instalação da ferramenta. A métrica da necessidade dos pacotes adicionais foi escolhida pelo fato de que pacotes adicionais influenciam diretamente no tempo de instalação da ferramenta, além de impactarem no desempenho do hardware da estação de gerência. Os dados foram obtidos pelo terminal da estação de gerência após o comando de

instalação da ferramenta e estão expostos no Quadro 5, da Seção 5.1.2 do Capítulo 5. Foi considerado que quanto menor a quantidade de pacotes, melhor a posição da ferramenta;

- Tempo de instalação: métrica quantitativa, porém subjetiva, tendo em vista que inclui o tempo para eventuais pesquisas com erros ocorridos durante a instalação e, ainda, o tempo necessário para a execução dos comandos seguindo a documentação oficial da ferramenta. O tempo de instalação foi considerado desde o momento em que o comando de instalação foi executado no terminal até o momento em que a ferramenta pôde ser acessada sem erros. Deve-se ressaltar que esse valor depende de alguns fatores, sendo um dos principais a taxa de *download* que a estação de gerência possui, neste caso, 100Mbps. O tempo de instalação de cada ferramenta pode ser visto no Quadro 6, da Seção 5.1.3, do Capítulo 5 e foi obtido através de cronometragem realizada pelo autor. Considera-se que quanto menor o tempo, melhor foi a colocação da ferramenta.

4.2.2.2 Métricas de configuração

As métricas desta subseção dizem respeito a configuração dos *hosts* e serviços que foram monitorados durante a análise.

- Autodescoberta de dispositivos: algumas ferramentas de gerenciamento possuem um recurso de autodescoberta de dispositivos para facilitar a configuração dos dispositivos da rede a serem monitorados. Essa métrica objetiva/qualitativa avaliou com “**sim**” ou “**não**” para sinalizar se a ferramenta possui o referido recurso. Considerou-se que ferramentas que receberam “sim” obtiveram melhor colocação. O resultado pode ser visto no Quadro 7, na Seção 5.2.1 do Capítulo 5;
- Criação de regras de monitoramento: a criação de regras permite que o administrador da rede personalize o monitoramento de acordo com suas preferências e com as características de determinados *hosts* ou serviços que estão sendo monitorados. As ferramentas receberam “**sim**” ou “**não**”, de acordo com a disponibilidade deste recurso, sendo que as ferramentas

que obtiveram “sim” ficaram melhor colocadas. Essa métrica é classificada como objetiva/qualitativa e seu resultado pode ser visualizado no Quadro 8, na Seção 5.2.2 do Capítulo 5;

- Agente próprio: para monitoramento de máquinas que não possuam o protocolo SNMP de forma nativa, ou às quais o administrador queira monitorar outros serviços, é necessária a instalação de um agente próprio da ferramenta. Essa métrica subjetiva/qualitativa sinalizou com “**sim**” as ferramentas que possuem agente próprio e quando o autor conseguiu instalar o mesmo na estação de gerência, e com “**não**” as ferramentas que não possuem ou quando o autor a encontrou, mas não foi possível realizar sua instalação na estação de gerência. As ferramentas marcadas com “sim” tiveram colocação superior e o resultado pode ser visualizado no Quadro 9, na Seção 5.2.3 do Capítulo 5;
- Grau de dificuldade: métrica subjetiva/qualitativa que expõe a dificuldade do autor com a configuração dos *hosts* e serviços a serem monitorados, de acordo com os seguintes critérios: **grau I**, para as ferramentas que possuam uma interface sugestiva o suficiente para que o autor conseguisse fazer a configuração dos *hosts*, sem precisar realizar pesquisas na documentação oficial ou em outros meios; **grau II**, para as ferramentas que exigiram a utilização da respectiva documentação oficial para a configuração de gerenciamento dos *hosts*; e as ferramentas receberam **grau III** quando a configuração dos *hosts* exigiu pesquisa extra documentação oficial, exigindo maior tempo por parte do autor, consequentemente gerando maior dificuldade. Quanto menor o grau de dificuldade da ferramenta, melhor foi a sua colocação neste quesito. O resultado pode ser visto no Quadro 10, na Seção 5.2.4 do Capítulo 5.

4.2.2.3 Métricas de informações

Um dos principais objetivos de uma ferramenta de gerenciamento é a geração de informações claras ao administrador da rede para que ele possa tomar decisões a respeito da rede gerenciada. Esta subseção expõe as métricas às quais

as ferramentas avaliadas durante este trabalho foram submetidas dentro do critério de geração de informações claras e objetivas.

- Mapas e/ou diagramas de rede: um importante recurso de uma ferramenta de gerenciamento e monitoramento é a capacidade da criação de mapas e/ou diagramas da rede gerenciada. Essa métrica visa classificar as ferramentas de forma subjetiva/qualitativa, indicando-as como: **boa** para as ferramentas que conseguem gerar mapas e/ou diagramas automaticamente e que ainda permitam a criação de mapas e/ou diagramas personalizáveis; **regular** para aquelas que possuem a opção de criação de mapas, porém não possuem a geração automática dos mesmos, considerando-se que este é um recurso importante, principalmente para redes com grande quantidade de dispositivos; e **ruim** para as que não possuem este recurso de forma nativa. Quanto melhor a classificação da ferramenta, melhor sua colocação. O resultado da avaliação desta métrica pode ser visualizado no Quadro 11, na Seção 5.3.1 do Capítulo 5;
- Geração de gráficos: é fundamental que as ferramentas de gerenciamento e monitoramento provenham ao administrador da rede gráficos objetivos e de fácil interpretação. Esta métrica foi definida pelos seguintes parâmetros de forma subjetiva/qualitativa: receberam qualificação **boa** as ferramentas que possuam gráficos intuitivos e personalizáveis (unidade de informação, datas, horários...); **regular** as que não possuem personalização ou boa apresentação dos dados; as ferramentas que não apresentam seus gráficos de forma clara e intuitivos, e nem permitem a personalização dos mesmos, foram classificadas como **ruim**. O resultado da avaliação desta métrica pode ser visualizado no Quadro 12, na Seção 5.3.2 do Capítulo 5, sendo que as ferramentas que obtiveram melhor qualificação, ficaram melhor posicionadas;
- Notificações via *e-mail*: O recurso de notificações permite que a ferramenta de gerenciamento envie *e-mails* com notificações em tempo real a respeito de eventos que ocorram nos elementos gerenciados. Neste trabalho, as ferramentas foram classificadas de forma objetiva e qualitativa, com “**sim**” ou “**não**” para sinalizar se possuem este recurso de

forma nativa. Ferramentas que obtiveram “sim”, ficaram com uma melhor colocação. O resultado pode ser visto no Quadro 13, na Seção 5.3.3 do Capítulo 5.

5 AVALIAÇÕES E RESULTADO

Esta Seção apresenta o resultado das avaliações de cada uma das ferramentas analisadas. As ferramentas foram avaliadas individualmente durante um período de sete dias.

Os comandos utilizados durante a instalação, configuração, ou outros comandos úteis que foram executados encontram-se nos apêndices deste trabalho. As avaliações encontram-se agrupadas pelo critério avaliativo e suas métricas, respectivamente.

5.1 INSTALAÇÃO DAS FERRAMENTAS

Encontram-se aqui as avaliações que dizem respeito a instalação das ferramentas e suas respectivas métricas de avaliação. As métricas foram previamente definidas e estão expostas na Seção 4.2.2.1.

5.1.1. Grau de dificuldade

Essa métrica remete as dificuldades encontradas pelo autor durante a instalação das ferramentas. Mais detalhes podem ser vistos na Seção 4.2.2.1, no item “grau de dificuldade”.

5.1.1.1 OMD

O OMD possui comandos fáceis e intuitivos para sua instalação, podendo todos ser obtidos no site oficial da ferramenta. Segue abaixo o passo a passo utilizado para a instalação da ferramenta, em sua versão 1.30, na estação de gerência supracitada:

1. *Download* do pacote da ferramenta – O primeiro passo para a instalação do OMD foi o *download* do pacote da ferramenta. As versões da ferramenta são obtidas no site oficial, no menu “*Download*”. Dentro da

página dos *downloads* há um *link* para “OMD *Releases”,* onde se encontra um diretório contendo pastas organizadas por distribuições Unix (Suse, Debian_Ubuntu, Centos_Rhel), além de uma pasta para a versão 1.30. Acessando a pasta “1.30”, pode-se visualizar todos os pacotes disponíveis nesta versão. Foi efetuado o *download* do pacote com extensão “.rpm”, que corresponde ao sistema operacional utilizado na estação de gerência (CentOS 7). O pacote possui um tamanho total de 87Mb;

2. Com o *download* do pacote concluído, foi iniciada a instalação do mesmo. Para isso, o terminal do sistema operacional foi aberto e navegou-se através das pastas até o local do arquivo em que estava o pacote baixado. Em seguida, executou-se o comando para instalação do pacote; este comando pode ser visualizado no Apêndice A;
3. Após aceitar as dependências de pacotes que a ferramenta exige, deu-se início a instalação do pacote principal;
4. Ao término da instalação do pacote principal, a ferramenta já estava instalada e os comandos para criação e edição dos *sites* já era possível;
5. Para criar um *site* chamado “tcc” e iniciá-lo foram executados os comandos do Apêndice B;

Segundo a documentação oficial, após esses passos, já deveria ser possível acessar a interface da ferramenta através do navegador da estação de gerência, porém, ao acessar o endereço *http://localhost/tcc*, o navegador retornou o erro 503 (Serviço Temporariamente Indisponível).

Para solucionar esse problema, efetuou-se uma pesquisa no FAQ (*Frequently Asked Questions*) que há no site oficial do OMD e descobriu-se que o erro está relacionado ao SELinux (*Security Enhaced Linux*). Segundo Hertzog e Mas (2015, p. 397), o “SELinux é um sistema de controle de acesso obrigatório. Na prática, o *kernel* consulta o SELinux antes de cada chamada do sistema para saber se o processo está autorizado a fazer a operação dada”. Foi realizada, então, a permissão de conexões de rede para o Apache com o comando contido no Apêndice C.

Resolvido o problema com o SELinux, novamente foi acessado o endereço *http://localhost/tcc* e, em seguida, o OMD solicitou a autenticação. Por padrão, o usuário é “omdadmin” e a senha “omd”, no entanto, após inserir os dados e efetuar o

login, o navegador retornou outro erro, dessa vez o erro 500 (*Internal Server Error*). Após diversas pesquisas, foi descoberto que o problema está relacionado com o arquivo “hashlib.py” (um módulo que, segundo a documentação oficial do *Python*, implementa algoritmos de criptografia *hash* e *message digest*) contido no diretório do OMD. A solução para o problema foi encontrada ao copiar o arquivo “hashlib.py” do diretório de instalação do Python para dentro do diretório do OMD; o comando utilizado pode ser visto no Apêndice D.

Acessando o endereço padrão do OMD após reiniciar o serviço, pôde-se visualizar a tela de boas-vindas da ferramenta. O OMD permite que o administrador escolha qual interface deseja utilizar, podendo ser a interface clássica do Nagios, o Check_MK, Thruk, Icinga ou o PNP4Nagios. Além das interfaces, a tela de boas-vindas ainda apresenta uma Wiki sobre o OMD.

Ao selecionar a opção de utilizar a interface do “Check_MK Multisite”, o navegador já foi redirecionado à página principal do Check_MK. A página inicial pode ser visualizada na Figura 7.

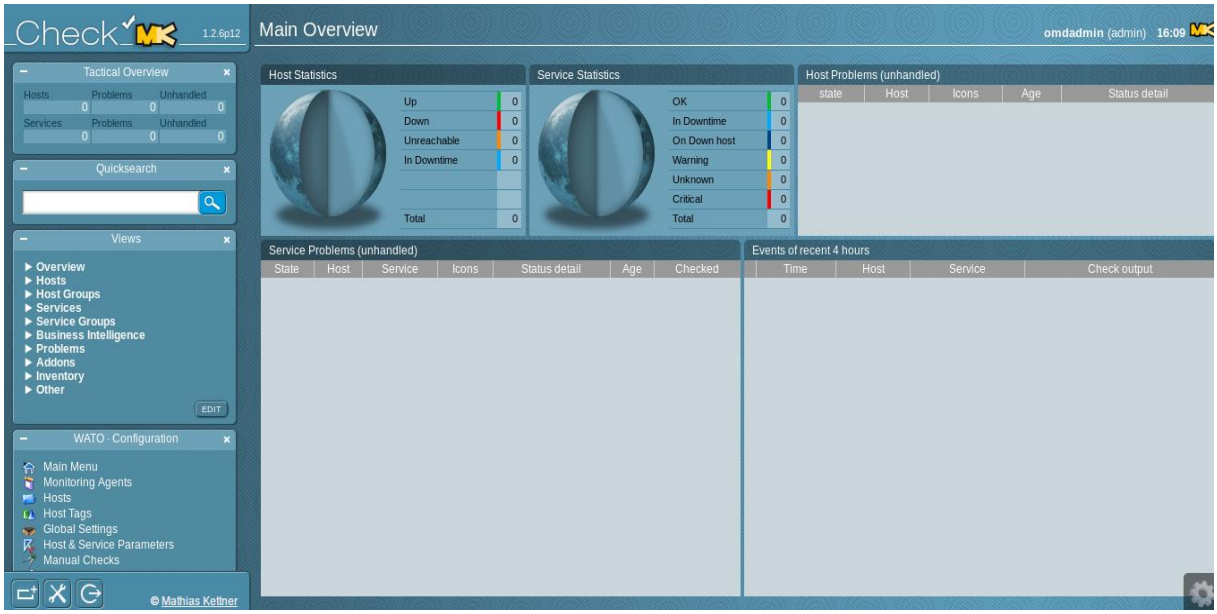


Figura 7 - OMD: Página inicial
Fonte: Autoria própria

Após esses passos, concluiu-se que a instalação do OMD estava finalizada. Levando em consideração os critérios previamente estabelecidos dentro da respectiva métrica, atribuiu-se à ferramenta uma dificuldade de **grau II**, pelo fato de que os passos contidos apenas na documentação da ferramenta não resultaram em seu pleno funcionamento, exigindo do autor a realização de pesquisas externas para

a solução de um dos problemas que ocorreram. Pelo grau de dificuldade atribuído à ferramenta, a mesma ocupou a segunda colocação nesta métrica.

5.1.1.2 Cacti

A primeira tentativa de instalação do Cacti foi através do passo a passo obtido na documentação oficial da ferramenta, porém, ao final da instalação o resultado esperado não foi obtido, já que a ferramenta não gerava nenhum tipo de gráfico. Por este motivo, seguiu-se tutoriais encontrados em pesquisas além da documentação da ferramenta. Também vale ressaltar que a instalação do Cacti requer mais conhecimentos em comandos Linux do que as outras ferramentas analisadas.

A seguir encontra-se o passo a passo efetuado para a instalação do Cacti na versão 0.8.8h.

1. O primeiro passo para a instalação do Cacti, é a instalação do Apache (servidor web), do PHP (linguagem de programação *back-end*), do MariaDB (servidor de banco de dados utilizado para armazenar as informações), do rrdtool (para a geração dos gráficos) e do net-snmp (um conjunto de ferramentas que operam com o protocolo SNMP). O comando utilizado para a instalação dessas ferramentas pode ser visualizado no Apêndice F;
2. Após a instalação dos referidos pacotes, foram iniciados e configurados para iniciar com o sistema os serviços do Apache, do MariaDB e do snmpd. Essa definição foi realizada através do comando contido do Apêndice G;
3. O terceiro passo a ser executado foi a criação do banco de dados para que o Cacti armazene os dados obtidos. Para isso, foi criada uma senha para o usuário do banco de dados e criado o banco de dados chamado “cacti”. Os comandos utilizados para isso podem ser visualizados no Apêndice H;
4. Com os passos anteriores concluídos, o pacote principal do Cacti já poderia ser instalado através do comando contido no Apêndice I;

5. A próxima etapa foi a configuração do banco de dados. Quando da instalação do pacote do cacti, é gerado um arquivo .sql que deve ser importado para o banco criado anteriormente através do comando contido no Apêndice J. Também deve ser editado o arquivo db.php alocado em /etc/cacti, de acordo com as configurações inseridas durante a criação do banco de dados;
6. Além disso, foi aberta uma porta no *firewall* para o Apache, com o comando contido no Apêndice K e configurado o *cron* (responsável pelo agendamento de tarefas em sistemas Linux) através do arquivo /etc/cron.d/cacti.

Com os referidos passos, a instalação do Cacti pôde ser finalizada acessando o endereço *http://localhost/cacti*. A página apresenta apenas a confirmação das configurações e requer a atualização da senha de acesso para o usuário *admin*. Feito isso, o Cacti está instalado e pronto para gerenciar os dispositivos da rede. A Figura 8 apresenta a página inicial do Cacti.

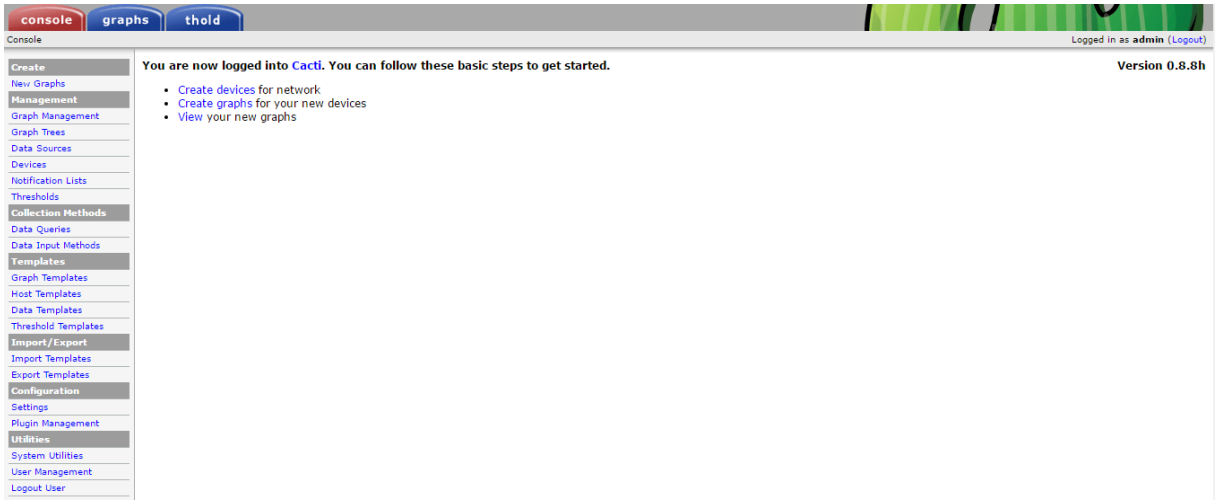


Figura 8 - Cacti: Página inicial
Fonte: Autoria própria

Pelos fatos apresentados, a referida ferramenta obteve **grau III** como resultado da avaliação sob o grau de dificuldade de instalação. Foi aplicado este resultado, uma vez que foi necessária a utilização de tutoriais extraoficiais para a instalação, além da instalação exigir do autor maiores conhecimentos técnicos, como a criação e configuração do banco de dados e configuração do *firewall* do sistema operacional. Este resultado posicionou a ferramenta na terceira colocação na referida métrica.

5.1.1.3 Zabbix

A documentação oficial do Zabbix possui diversos tutoriais para a instalação da ferramenta, com os métodos descritos passo a passo para os três tipos de instalação possíveis: instalação através de pacotes provenientes dos repositórios, instalação através dos fontes e compilação pelo usuário, ou instalação através dos *containers*. Para o desenvolvimento deste trabalho, foi utilizada a instalação através do repositório.

Após a atualização do sistema operacional, foram executados os seguintes comandos para a instalação da ferramenta Zabbix em sua versão 3.0.5:

1. O Zabbix exige que a estação de gerência possua o pacote LAMP (Linux, Apache, MySQL, PHP) instalado. Para instalar e configurar essas ferramentas foram utilizados os comandos contidos no Apêndice L. A instalação desses pacotes obteve sucesso logo na primeira tentativa, sem maiores dificuldades;
2. Após a instalação do LAMP, o Zabbix foi instalado através do repositório com o comando apresentado no Apêndice M. A instalação do pacote principal do Zabbix também ocorreu sem problemas;
3. Com a conclusão da instalação do pacote do Zabbix, foram instalados os pacotes necessários para a integração com o banco de dados MySQL e também o agente utilizado para monitorar a própria estação de gerência. Os comandos utilizados são listados no Apêndice N;
4. O próximo passo exigido foi a configuração do banco de dados MySQL e a cópia dos dados e esquemas iniciais para o banco de dados recém-criado. Os comandos utilizados neste processo estão presentes no Apêndice O.

Com a conclusão dos passos supracitados, a interface do Zabbix já estava acessível através do endereço *http://localhost/zabbix,* no primeiro acesso, é apresentada uma verificação da instalação e as últimas configurações necessárias.

A Figura 9 apresenta a tela de confirmação das configurações, e a Figura 10 apresenta as últimas configurações necessárias.

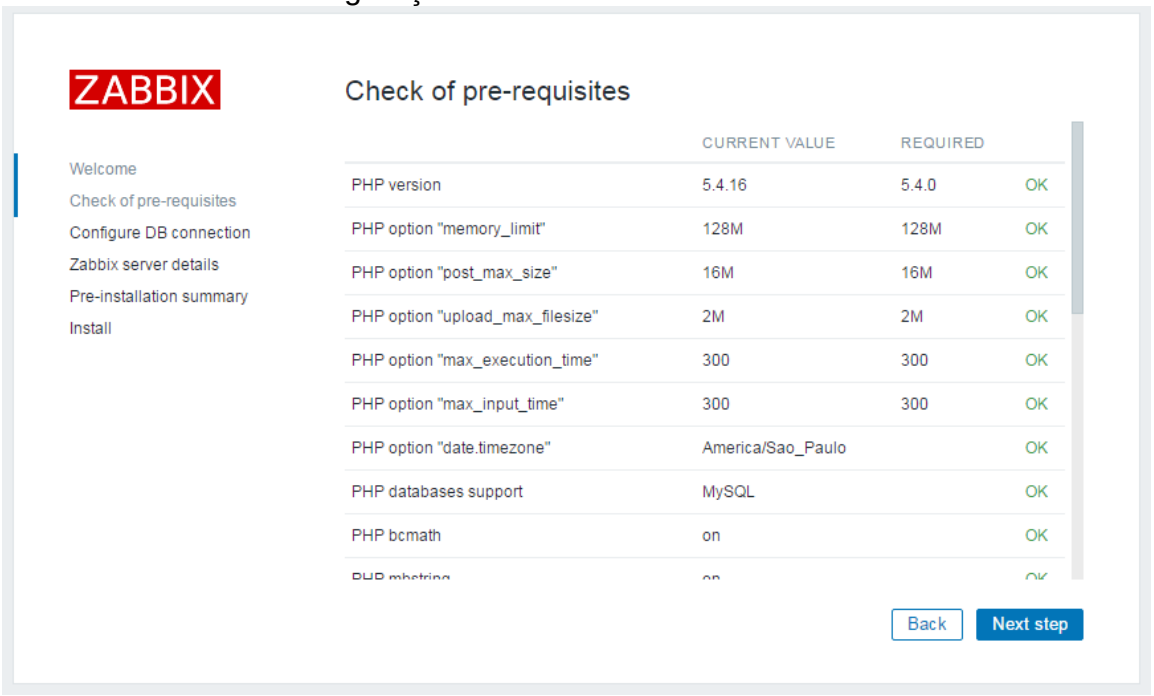


Figura 9 - Zabbix: Verificação da instalação
Fonte: Autoria própria

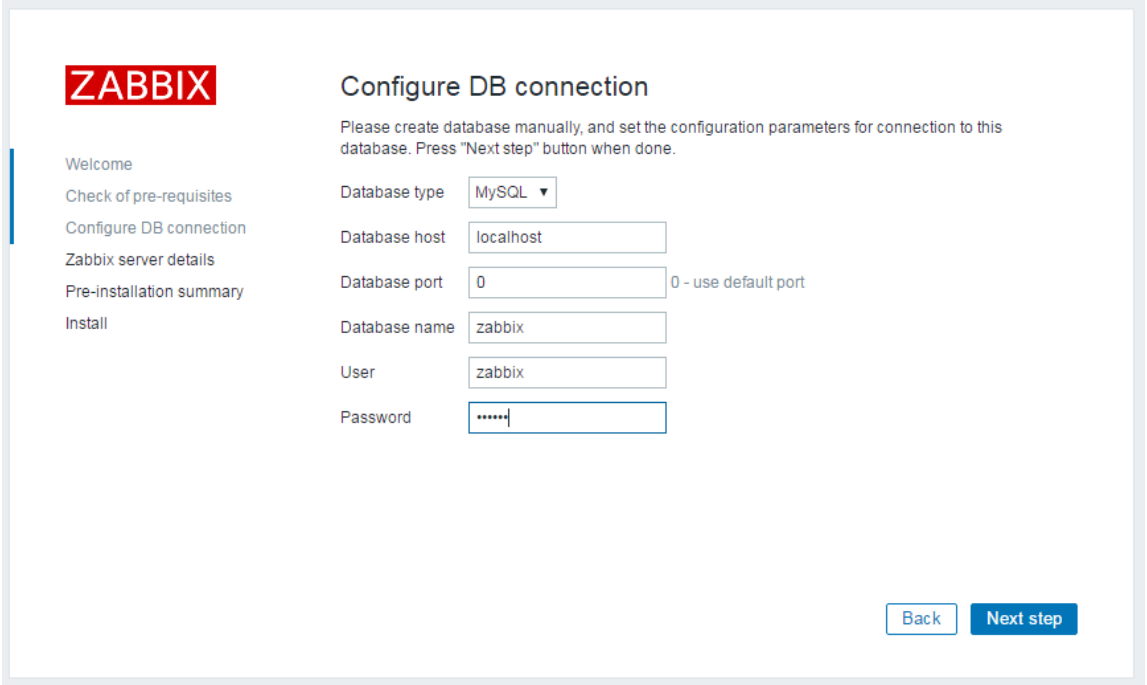


Figura 10 - Zabbix: Configuração do banco de dados
Fonte: Autoria própria

Após a inserção dessas informações finais, o Zabbix exigiu o usuário e senha para acesso (o padrão é admin/zabbix); inserindo o usuário e senha, o *dashboard* foi exibido, conforme mostra a Figura 11.

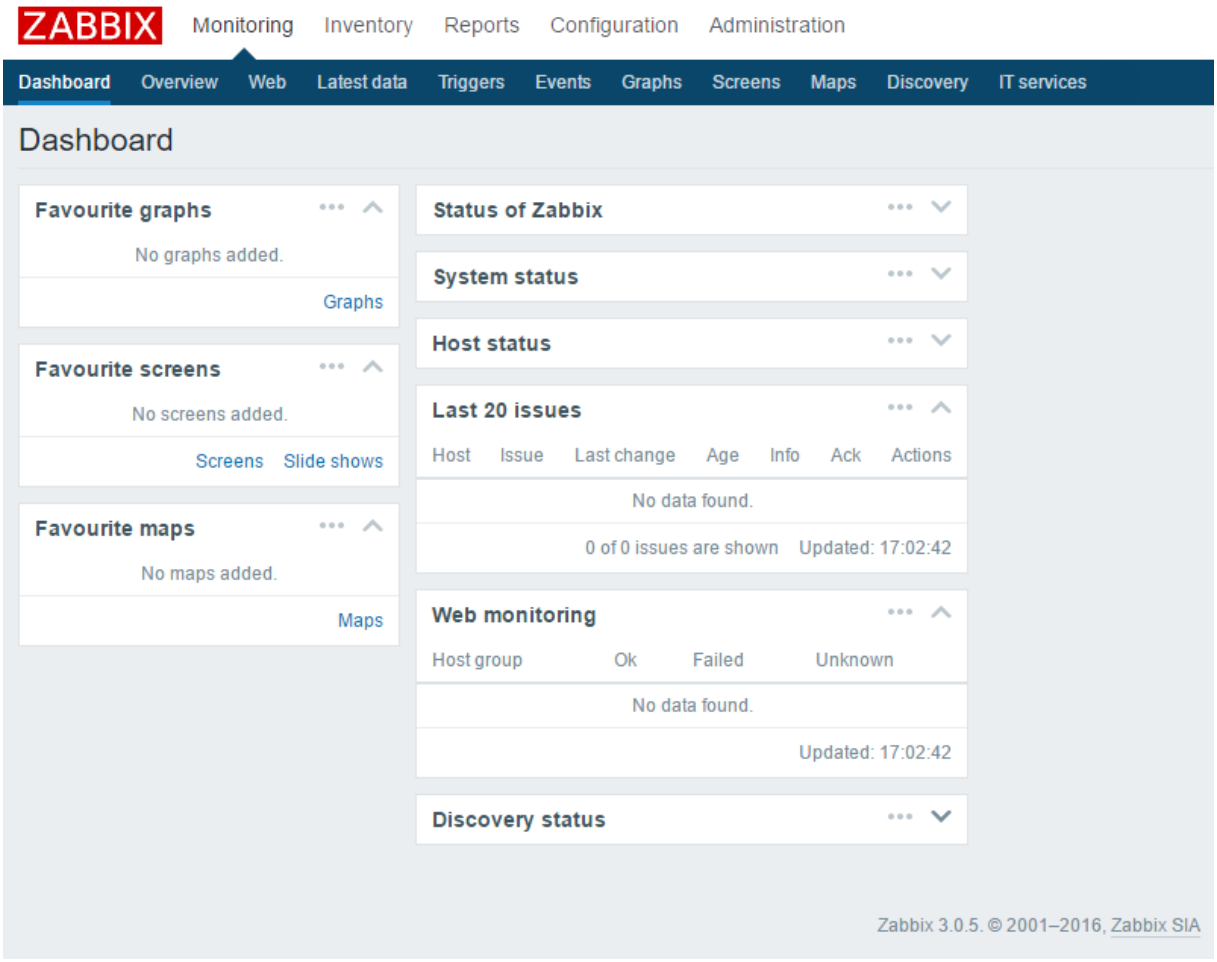


Figura 11 - Zabbix: Página inicial
Fonte: Autoria própria

Pelos fatos apresentados durante esta Seção, atribuiu-se à ferramenta Zabbix o **grau I**, já que mesmo necessitando a configuração do banco de dados, todos os passos foram obtidos da própria documentação oficial. Por este motivo, o Zabbix ficou com a primeira colocação nesta métrica avaliativa.

O Quadro 4 resume o resultado da avaliação da métrica em questão.

Ferramenta	Grau de dificuldade	Posição da Ferramenta
Zabbix	Grau I	1ª colocação
OMD	Grau II	2ª colocação
Cacti	Grau III	3ª colocação

Quadro 4 - Resultado: Grau de dificuldade de instalação
Fonte: Autoria própria

5.1.2 Necessidades de pacotes adicionais

Nesta Seção são apresentados os resultados obtidos durante a avaliação, que dizem respeito a referida métrica, previamente estabelecida no item de mesmo nome, na Seção 4.2.2.1 deste trabalho.

5.1.2.1 OMD

O OMD necessitou **61 pacotes** dependentes para concluir a instalação, com um tamanho total de 326Mb, sendo que 42Mb foram baixados dos repositórios em questão e ocuparam 474Mb após a instalação. Pela quantidade de pacotes necessários, o OMD ocupou a terceira colocação nesta métrica.

5.1.2.2 Cacti

Diferentemente do OMD, o Cacti requer que outras ferramentas sejam previamente instaladas, por isso, foram contabilizados os pacotes dependentes de todos os passos da instalação.

Previamente ao Cacti, foram instalados 15 pacotes com mais 32 dependências. Soma-se a esses pacotes, o pacote principal do Cacti, resultando em **48 pacotes** instalados, com 39,5Mb baixados e ocupando 170,4Mb em disco após a instalação. Com a quantidade de pacotes expostos, o Cacti ocupou a segunda colocação neste quesito.

5.1.2.3. Zabbix

Assim como o Cacti, o Zabbix também exige que sejam instaladas outras ferramentas previamente, neste caso, o LAMP. O LAMP instalou 4 pacotes com 19 dependências, totalizando 29,3Mb de *download* e ocupando 139Mb em disco depois de instalado. Após a instalação do LAMP, o Zabbix instalou outros 4 pacotes com 12 dependências, com 8,6Mb baixados do repositório e 41Mb após a instalação. Quando todos os pacotes já haviam sido instalados, totalizou-se **39 pacotes**, com

37,9Mb baixados e ocupando 180Mb em disco. Com este resultado, o Zabbix ocupou a primeira colocação no que se refere a quantidade de pacotes necessário.

O Quadro 5 resume o resultado da avaliação da métrica em questão.

Ferramenta	**Quantidade de Pacotes**	**Posição da Ferramenta**
Zabbix	39 pacotes	1ª colocação
Cacti	48 pacotes	2ª colocação
OMD	61 pacotes	3ª colocação

Quadro 5 - Resultado: Quantidade de pacotes adicionais
Fonte: Autoria própria

5.1.3 Tempo de instalação

Na Seção a seguir encontram-se as avaliações e resultados da referida métrica, de acordo com os requisitos previamente estabelecidos na Seção 4.2.2.1.

5.1.3.1 OMD

A instalação do pacote principal do OMD, juntamente com as suas dependências, demorou aproximadamente 4 minutos, porém, como citado anteriormente, aconteceram alguns erros durante a instalação da ferramenta. Para a resolução desses erros foram necessárias algumas pesquisas para buscar a solução, pesquisas essas que aumentaram consideravelmente o tempo da instalação, concluída em aproximadamente **28 minutos**. Esse tempo atribuiu ao OMD a terceira posição neste quesito.

5.1.3.2 Cacti

Como previamente dito, a instalação seguindo os passos apresentados na documentação oficial da ferramenta não acarretou no seu perfeito funcionamento, por este motivo, o tempo gasto durante a primeira instalação não foi considerado nesta avaliação. Apesar de requerer maiores configurações, a instalação do Cacti foi relativamente rápida, durando aproximadamente **24 minutos**, sendo que a maior

parte deste tempo foi usado para a configuração do banco de dados, do *firewall* e do Apache. O Cacti obteve a segunda colocação neste quesito.

5.1.3.3 Zabbix

O Zabbix foi a única ferramenta testada em que não aconteceram erros durante a instalação, sendo que todo o passo a passo para a instalação foi retirado da documentação oficial da ferramenta. A instalação da ferramenta, incluindo as configurações necessárias para a conclusão da instalação foi realizada em aproximadamente **18 minutos**, o que pode ser considerada uma instalação rápida em comparação com as outras ferramentas testadas. O tempo de 18 minutos atribuiu ao Zabbix a primeira colocação neste quesito.

O Quadro 6 apresenta um resumo com os resultados obtidos desta métrica.

Ferramenta	**Tempo de Instalação**	**Posição da Ferramenta**
Zabbix	18 minutos	1ª colocação
Cacti	24 minutos	2ª colocação
OMD	28 minutos	3ª colocação

Quadro 6 - Resultado: Tempo de instalação
Fonte: Autoria própria

5.2 CONFIGURAÇÃO DE *HOSTS* E SERVIÇOS

Encontram-se aqui as avaliações que dizem respeito a configuração dos *hosts* e serviços que foram monitorados durante o período da análise e suas respectivas métricas de avaliação. As métricas foram previamente definidas e podem ser vistas na Seção 4.2.2.2.

5.2.1 Autodescoberta de dispositivos

Encontra-se nesta Seção as avaliações e resultados referentes a autodescoberta de dispositivos, métrica esta que foi previamente estabelecida na Seção 4.2.2.2 do Capítulo 4.

5.2.1.1 OMD

Como citado anteriormente, o OMD utiliza em seu *core* o Nagios e a interface do Check_MK. Mesmo já integrando diversos *plug-ins*, o OMD não possui a opção de autodescoberta de *hosts*, sendo que também não foi encontrado pelo autor nenhum *plug-in* com tal capacidade, portanto, deu-se o valor de "**não**" para tal métrica. Vale ressaltar que o Check_MK implementa o autodescoberta dos serviços que podem ser monitorados em determinado *host*, porém, esse recurso não foi analisado durante o trabalho. O OMD ficou na segunda colocação na referida métrica.

5.2.1.2 Cacti

De forma nativa, o Cacti **não** possui o recurso de autodescoberta de dispositivos, sendo necessário que o administrador da rede configure todos os dispositivos à serem gerenciados. Assim como outras ferramentas de gerenciamento, o Cacti também possui uma grande comunidade ativa de usuários e, com isso, diversos *plug-ins* podem ser obtidos no site oficial da ferramenta, sendo que o *plug-in* para este recurso pode ser facilmente obtido, no entanto esta opção não foi considerada neste trabalho. Com este resultado, o Cacti também ocupou a segunda colocação.

5.2.1.3 Zabbix

O Zabbix é a única ferramenta analisada que possui de forma nativa o recurso de autodescoberta de *hosts*. Para o funcionamento deste recurso, é necessário que o administrador da rede configure uma "regra de descobrimento". Para a criação dessa regra acessou-se o menu "*configuration"*, em seguida o sub-menu "*Discovery*". Nesta tela são exibidas todas as regras de descobrimento já configuradas e também um botão no canto superior direito, "*Create discovery rule"*, para que seja criada uma nova regra.

Ao clicar no botão é apresentada a página para criação de uma nova regra, conforme mostra a Figura 12.

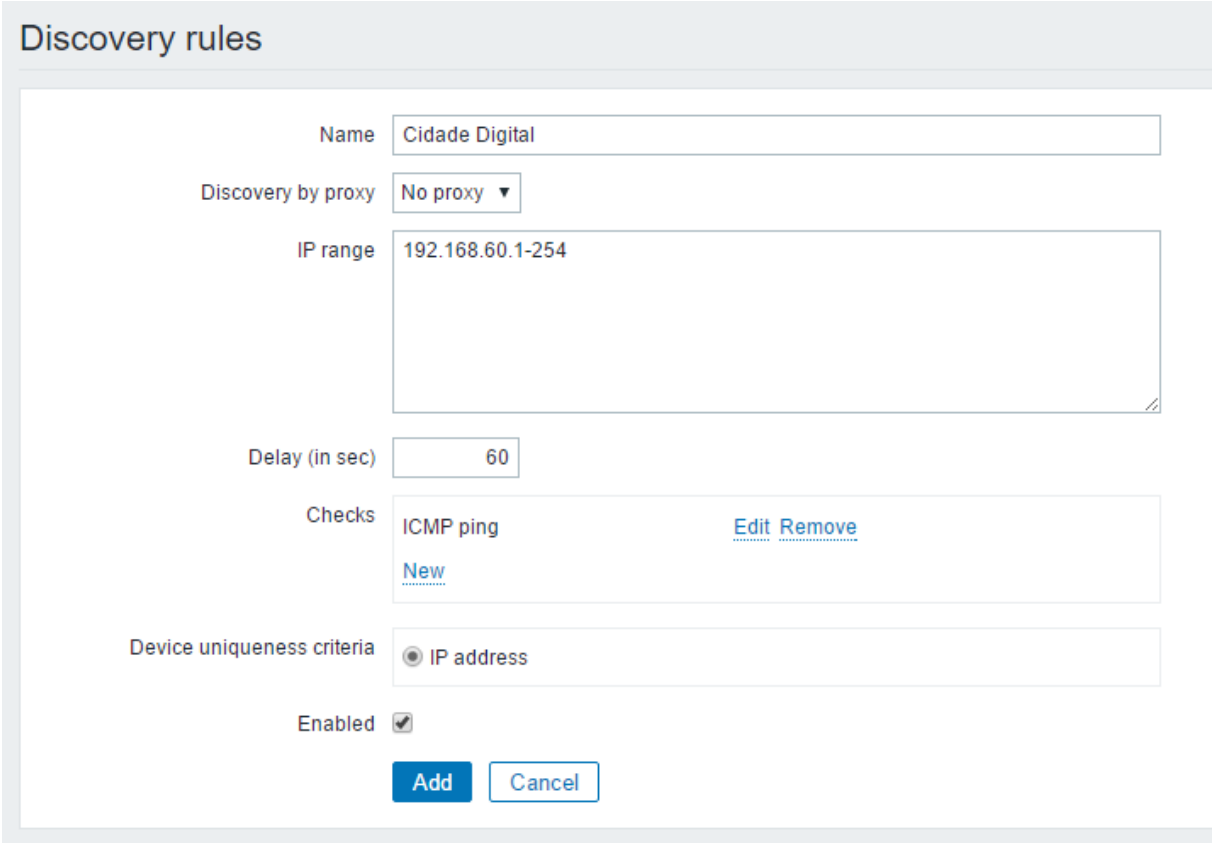


Figura 12 - Zabbix: Página para criação de uma nova regra de autodescoberta
Fonte: Autoria própria

As informações inseridas dizem respeito a regra criada, sendo:

- *Name*: nome dado à regra criada;
- *Discovery by proxy*: se será utilizado algum *proxy* para realizar a descoberta dos dispositivos na rede; neste caso, não há;
- *IP range*: faixa de IP's a qual o Zabbix deverá tentar descobrir os dispositivos, neste caso, o Zabbix irá buscar dispositivos do IP 192.168.60.1 até o IP 192.168.60.254;
- *Delay (in sec)*: intervalo de tempo em que o Zabbix fará as tentativas de descobrimento;
- *Checks*: tipo de checagem que o Zabbix irá utilizar para descobrir os *hosts*. Neste caso, foi utilizado o método ICMP Ping, mas outras opções estão disponíveis, como, por exemplo, o próprio agente Zabbix;
- *Device uniqueness criteria*: podendo ser pelo *check* ou o endereço IP, para os fins aqui propostos, foi utilizado o endereço IP;

- *Enabled*: deixa a regra ativada ou desativada após a configuração

Após adicionar a regra, o Zabbix já é capaz de encontrar os dispositivos. Para visualizar os dispositivos encontrados, acessa-se o menu "*Monitoring*" > "*Discovery*". A Figura 13 exibe todos os dispositivos encontrados pela autodescoberta.

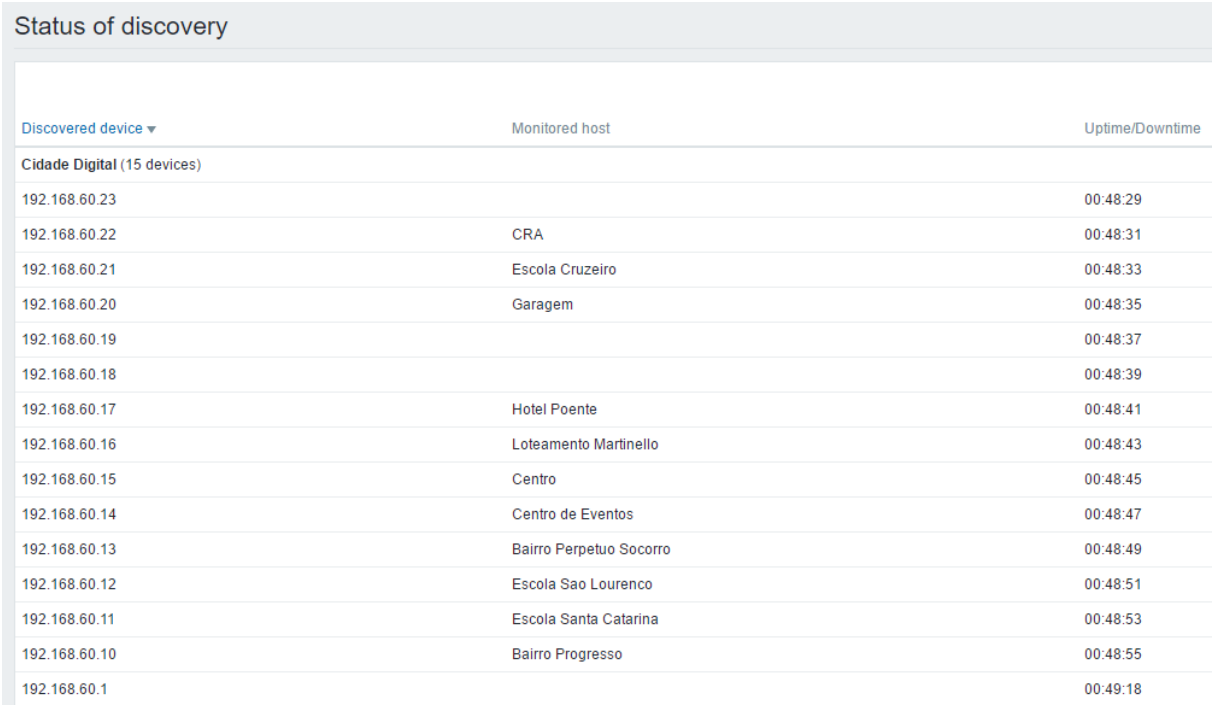


Figura 13 - Zabbix: Dispositivos encontrados pelo recurso de autodescoberta
Fonte: Autoria própria

Ao final desta etapa, o Zabbix encontrou os dispositivos seguindo a regra criada, entretanto, nenhuma ação foi executada com os *hosts* encontrados. Foram criadas, assim, ações para descobrimento, ou seja, quando o Zabbix encontrava um novo dispositivo, uma ação deveria ser executada, neste caso, a adição do *host* e a vinculação dele a um *hostgroup*.

Para criar uma ação, deve ser acessado menu “*Configuration*” > *“Actions”.* A Figura 14 exibe a primeira etapa da criação de uma *action*; nessa primeira etapa só deve ser inserido o nome da *action*, todas as outras opções já vêm inseridas.

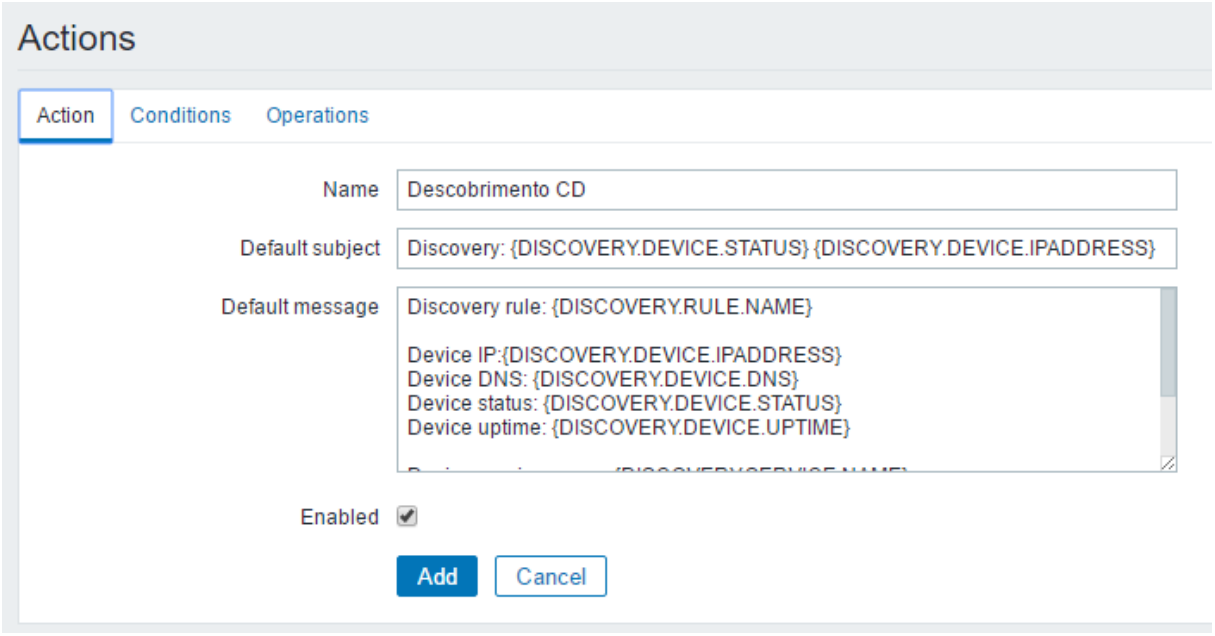


Figura 14 - Zabbix: Nova *action*
Fonte: Autoria própria

A segunda etapa é acessar a aba “*Conditions*”, e inserir as condições para a *action*. Aqui foram colocadas como condições que a descoberta seja da regra “Cidade Digital” e que o *check* seja do tipo ICMP Ping. A Figura 15 apresenta a tela dessas configurações.

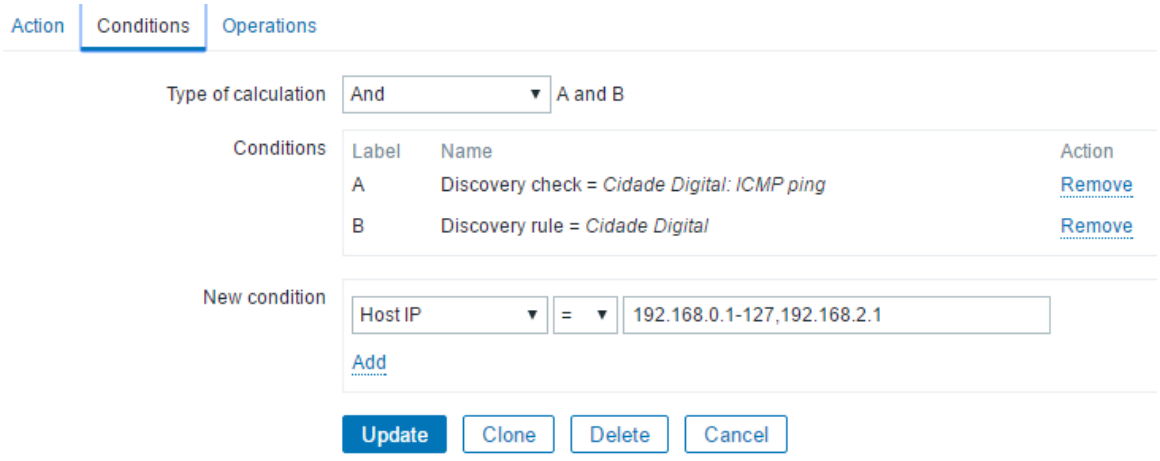


Figura 15 - Zabbix: Condições para uma *action*
Fonte: Autoria própria

A última etapa para a criação de uma *action* é configurar qual será a operação executada quando o Zabbix descobrir um novo dispositivo na rede. Neste

caso, foi configurado que os dispositivos encontrados serão adicionados aos *hosts* e que serão alocados no *hostgroup* Cidade Digital. A Figura 16 exibe essa configuração.

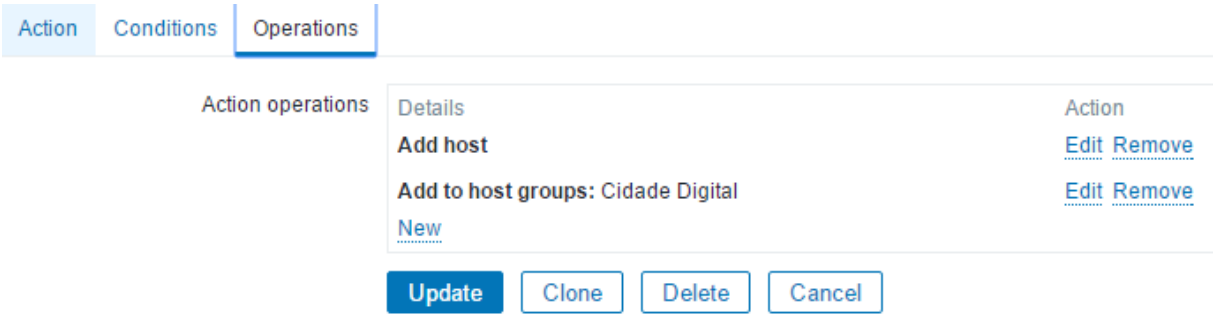


Figura 16 - Zabbix: Operações para uma *action*
Fonte: Autoria própria

Ao final da criação da *action*, basta aguardar o tempo configurado para o Zabbix fazer uma nova autodescoberta para que a *action* configurada seja executada. Pelos fatos apresentados durante este trabalho, atribuiu-se **sim** ao Zabbix no que compete a capacidade de autodescoberta de *hosts* na rede monitorada. Esse recurso é muito importante em uma ferramenta de monitoramento e gerenciamento, levando em conta que facilita o trabalho do administrador e reduz o tempo necessário para realizar a configuração da ferramenta, além do fato de que redes de computadores estão em constante crescimento e, com este recurso, novos dispositivos na rede são adicionados automaticamente à ferramenta, seguindo regras pré-estabelecidas. Pelo fato de ser a única ferramenta analisada que possui este recurso, o Zabbix ocupou a primeira colocação nesta métrica.

O Quadro 7 expõe o resultado da avaliação da referida métrica.

Ferramenta	**Recurso de autodescoberta de *hosts***	**Posição da Ferramenta**
Zabbix	Sim	1ª colocação
OMD	Não	2ª colocação
Cacti	Não	2ª colocação

Quadro 7 - Resultado: Autodescoberta de *hosts*
Fonte: Autoria própria

5.2.2 Criação de regras de monitoramento

Nesta Seção encontram-se as avaliações e resultados condizentes com a respectiva métrica exposta na Seção 4.2.2.2 deste trabalho.

5.2.2.1 OMD

Através do Check_MK, o OMD entrega ao administrador da rede uma interface intuitiva e de fácil manipulação. Por meio desta interface é que o administrador tem a possibilidade de criar regras tanto para os *hosts* gerenciados, quanto para seus serviços.

Um dos principais recursos que o Check_MK fornece é o WATO (*Web Administration Tool*), um menu que o administrador pode acessar diversas opções disponíveis na ferramenta: a criação de pastas para agrupar os *hosts* monitorados, configurações gerais da ferramenta, criação de usuários e regras de monitoramento, entre outros. A Figura 17 exibe este menu.

Figura 17 - OMD: WATO (*Web Administration Tool*)
Fonte: Autoria própria

As regras de monitoramento podem ser acessadas no item "*Host & Service Parameters*". Dentro deste item as possíveis regras são agrupadas em subitens, também há um campo de pesquisa para que o administrador possa buscar determinada regra que deseje aplicar.

A Figura 18 apresenta a página com as opções para criação de regras de monitoramento na ferramenta OMD.

Figura 18 - OMD: Regras de monitoramento
Fonte: Autoria própria

Durante o desenvolvimento do trabalho, as únicas regras criadas foram para estipular a velocidade das interfaces *wireless* das estações de rádio monitoradas e também para definir que todas as interfaces monitoradas teriam a unidade de medida *bits*. A regra foi definida pelos seguintes passos, a partir do menu de "*Parameters for discovered services*":

- Acessou-se o menu de "*Network interfaces and switch ports*", que remete às regras de monitoramento para as interfaces e portas dos *hosts* monitorados;
- Foi selecionada a pasta "*main*" para que todos os *hosts* monitorados recebessem a regra;
- Aplicou-se o parâmetro "*Measurement unit*" e foi selecionada a opção "*bits*".

A regra criada para alteração da unidade de medida ficou como indica a Figura 19.

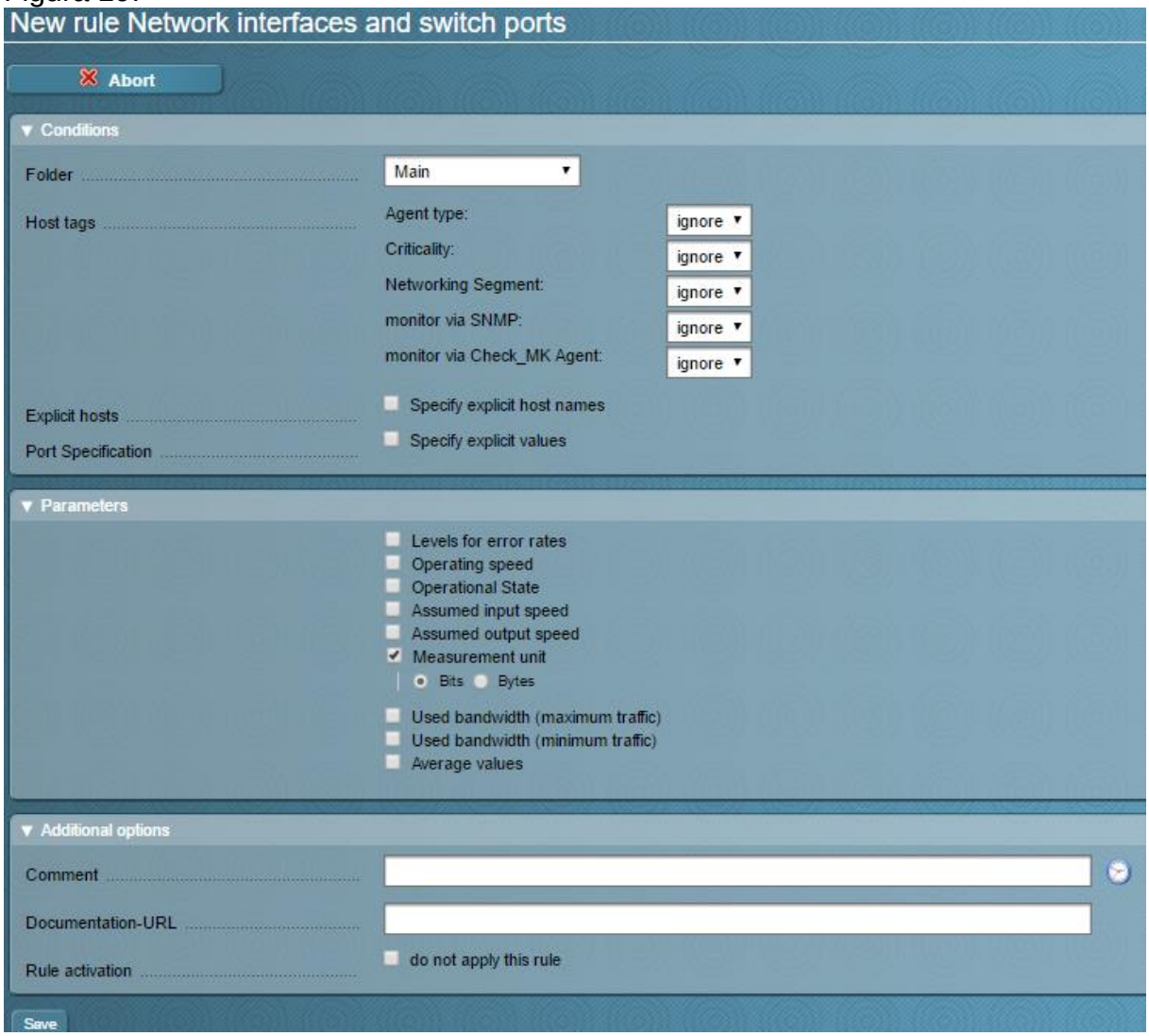


Figura 19 - OMD: Regras de monitoramento, unidade de medida
Fonte: Autoria própria

A regra para estipular a velocidade da interface *wireless* de cada estação monitorada em 54Mbps foi definida da mesma forma que a anterior, porém, ao invés de selecionarmos o parâmetro "*Measurement unit*", foi selecionado o parâmetro "*Assumed input speed*" e "*Assumed output speed*" e inserido o valor de 54Mbps correspondente em *bits* (56623104). Por especificarmos os *hosts* que receberiam tal regra e também a interface, a regra foi criada na pasta "*main*".

A Figura 20 apresenta a regra que determina que as interfaces *wireless* terão velocidade de 54Mbps.

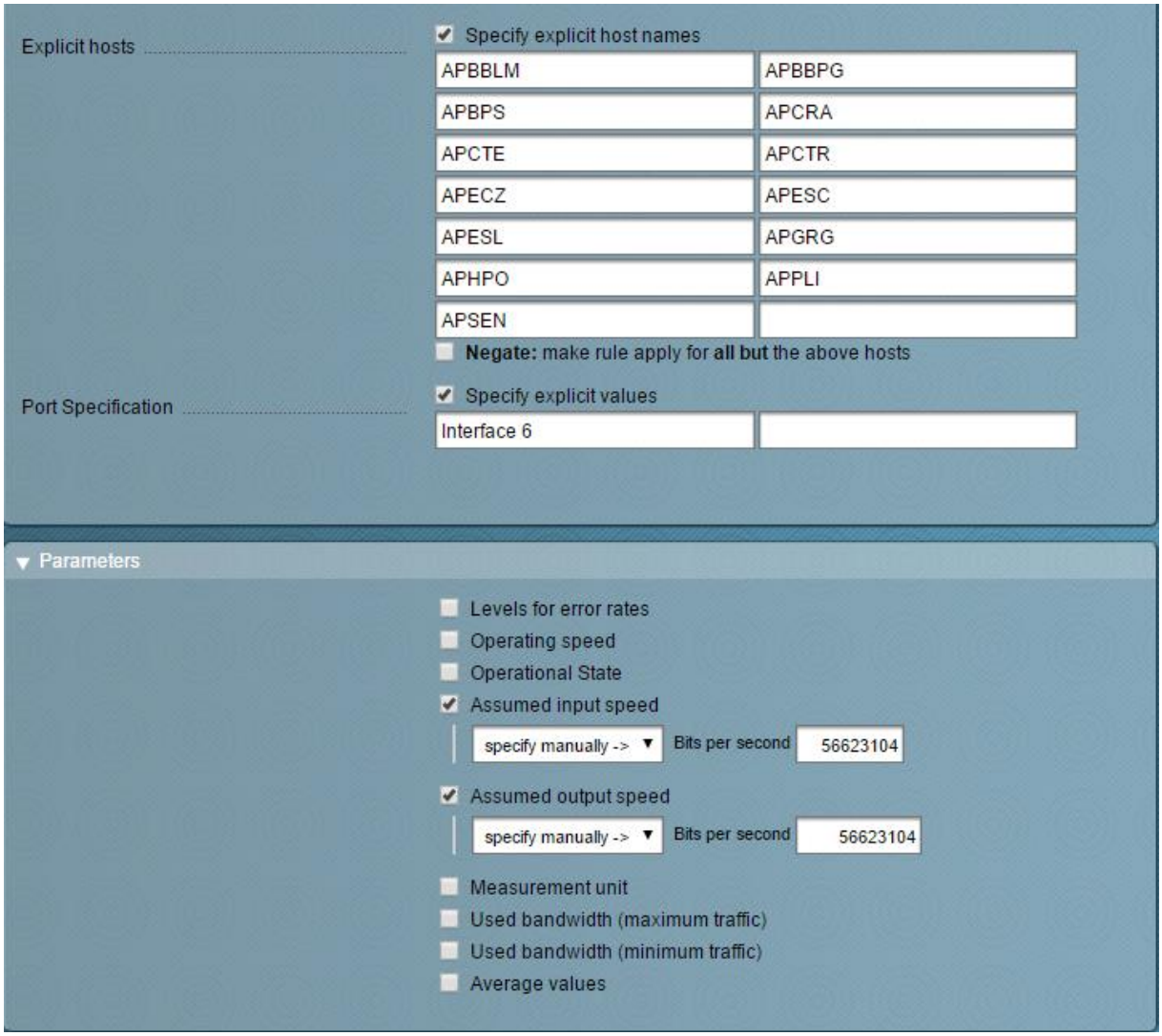


Figura 20 - OMD: Regras de monitoramento, velocidade das interfaces wireless
Fonte: Autoria própria

Essas foram as regras implementadas durante a análise, para a ferramenta OMD, mas vale ressaltar que a ferramenta possui diversos outros tipos de regras para que o administrador da rede personalize o monitoramento de seus *hosts.* Pelos fatos apresentados, deu-se para a ferramenta OMD a atribuição "**sim**" para a referida métrica, colocando-a na primeira posição neste quesito.

5.2.2.2 Cacti

Em sua instalação padrão e sem a adição de *plug-ins* providos pela comunidade, o Cacti mostra-se uma ferramenta voltada mais ao monitoramento de ativos do que propriamente o gerenciamento dos mesmos. O Cacti **não** apresenta nenhum recurso para a criação de regras de monitoramento, restringindo o monitoramento às opções padrões da ferramenta. Pelo fato apresentado, a ferramenta ocupou a segunda colocação.

5.2.2.3 Zabbix

O Zabbix não possui um menu explícito com as regras que podem ser criadas para o monitoramento, todavia, pode-se considerar que ele possui regras de monitoramento visto as opções que dispõe ao administrador da rede. Entre as opções que podem ser consideradas regras de monitoramento estão as *actions*, supracitadas na Seção anterior e utilizadas para adição dos *hosts* automaticamente após a descoberta dos mesmos pela função de autodescoberta.

Além das *actions*, o Zabbix ainda permite a personalização de todas as checagens que realiza e de suas *triggers*. As *triggers* são “expressões lógicas que analisam os dados coletados pelos itens e representam o estado do sistema” (ZABBIX, 2016). É através das *triggers*, por exemplo, que um *host* pode mudar seu *status* após ter o tempo de resposta maior que 20ms (milissegundos).

O Zabbix trabalha com *templates* vinculados aos *hosts. Template* é um conjunto de entidades (gráficos, *triggers*, aplicações e itens) que são vinculados aos *hosts*, ou seja, o administrador pode vincular todos os seus *hosts* ao *template* ICMP Ping para obter informações relativas ao tempo de resposta do dispositivo e outros *templates* específicos do dispositivo. Para modificar ou criar uma nova *trigger* deve-se acessar o menu “*Configuration*” > “*Templates*”. Será exibida uma página com todos os *templates* disponíveis e suas respectivas *triggers.* Clicando no link *“triggers”*, são apresentadas as *triggers* do *template*.

Para elucidar, a Figura 21 apresenta as *triggers* vinculadas ao *template* ICMP Ping.

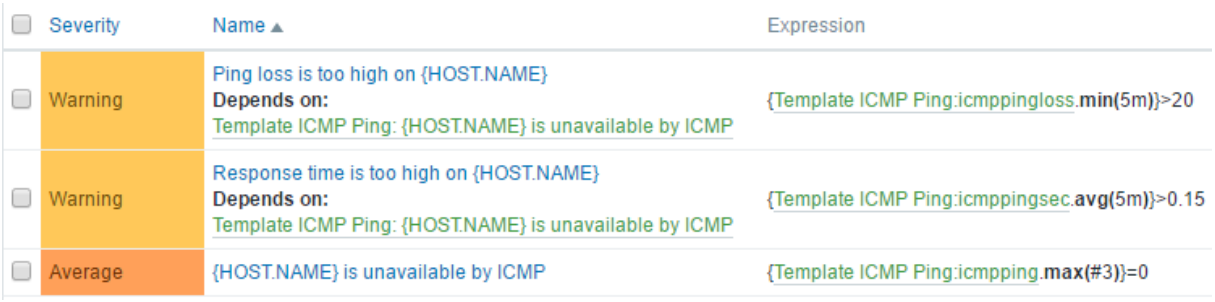


Figura 21 - Zabbix: *Triggers* do *template* ICMP Ping
Fonte: Autoria própria

A Figura apresenta três *triggers* que simbolizam os seguintes estados aos *hosts*:

- A primeira *trigger* apresenta um estado de “aviso” (*warning*) se a quantidade de pacotes perdidos for maior do que 20% em um período de no mínimo 5 minutos;
- A segunda também apresenta um estado de “aviso” se o tempo de resposta do dispositivo for maior que 0,15ms em um período médio de 5 minutos;
- E a última *trigger* apresenta um estado de “médio cuidado” (*average*) ao dispositivo quando as últimas três capturas de dados resultaram em uma resposta 0, ou seja, o dispositivo não respondeu ao protocolo.

Para fins de estudo, as três *triggers* foram modificadas e outra foi criada durante os testes, transformando-as em:

- Estado de “informação” (*information*) para quando a quantidade de pacotes perdidos for maior do que 15% em um período de no mínimo 5 minutos;
- Estado de “aviso” quando o tempo de resposta no dispositivo for maior que 10ms em uma média de 5 minutos;
- Estado de “alto cuidado” (*high*) quando as últimas três tentativas retornarem 0;
- Criada uma nova *trigger* para tratar como “estado de calamidade” (*disaster*) quando as últimas cinco tentativas retornarem 0.

A Figura 22 apresenta essas *triggers* modificadas.

Figura 22 - Zabbix: *Triggers* modificadas no *template* ICMP Ping
Fonte: Autoria própria

A personalização de *triggers* e criação de *actions* são apenas exemplos de regras indiretas que podem ser criadas no monitoramento de uma rede com o Zabbix, porém outras opções estão disponíveis, como a criação de *templates* únicos para determinados dispositivos. Por esses fatos apresentados, foi atribuído **sim** para o Zabbix no que se refere a métrica de criação de regras de monitoramento, colocando a ferramenta na primeira colocação juntamente com o OMD.

O Quadro 8 exibe o resultado desta métrica.

Ferramenta	**Criação de regras de monitoramento**	**Posição da Ferramenta**
OMD	Sim	1ª colocação
Zabbix	Sim	1ª colocação
Cacti	Não	2ª colocação

Quadro 8 - Resultado: Criação de regras de monitoramento
Fonte: Autoria própria

5.2.3 Agente próprio

A Seção a seguir expõe as avaliações e resultados de acordo com a métrica definida na Seção 4.2.2.2 deste trabalho.

5.2.3.1 OMD

Utilizando a interface do Check_MK para aproveitar grande parte dos recursos disponíveis no OMD, o agente do OMD acaba sendo o próprio agente do Check_MK. Como citado anteriormente, grande parte dos recursos e configurações

provenientes do Check_MK são obtidos no menu principal do WATO. O primeiro ícone que há neste menu é justamente o que dá acesso aos agentes do Check_MK. Acessando esta página, o administrador tem a sua disposição todos os agentes que a ferramenta concede, alguns estão exibidos na Figura 23.

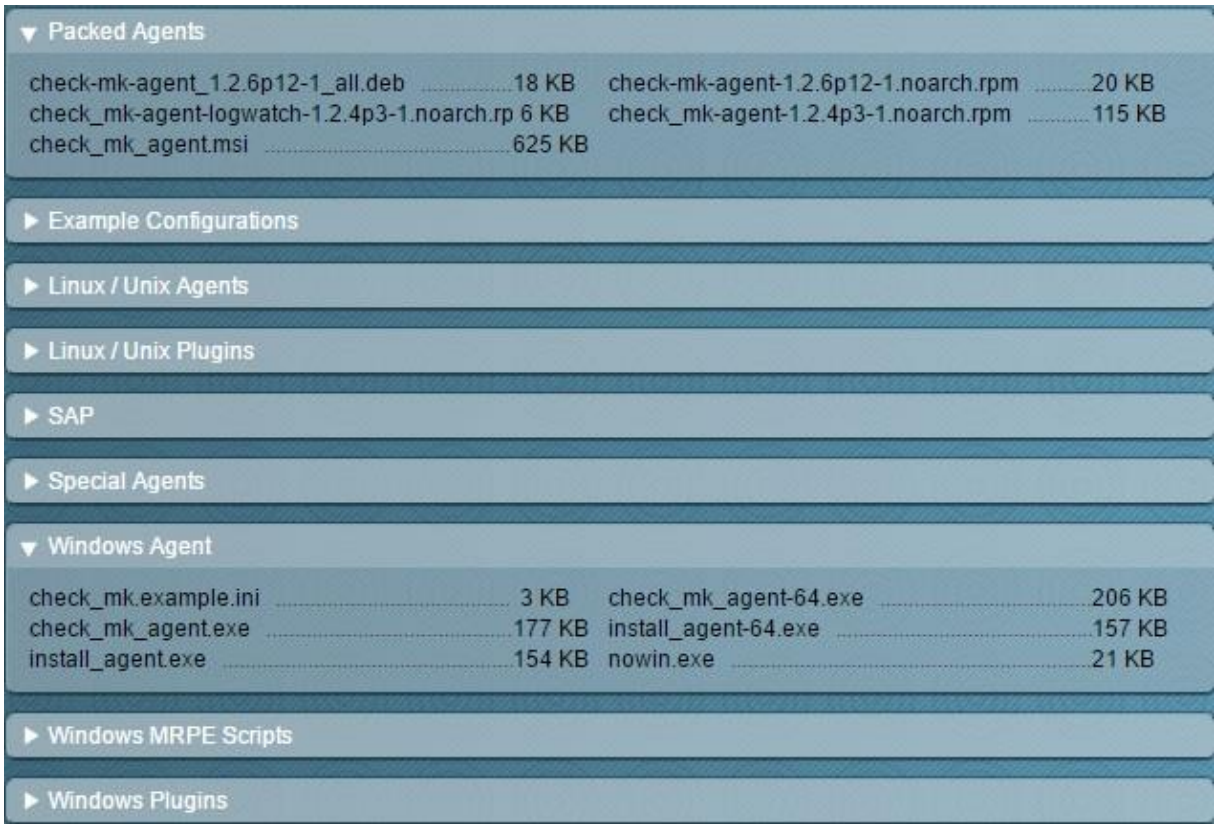
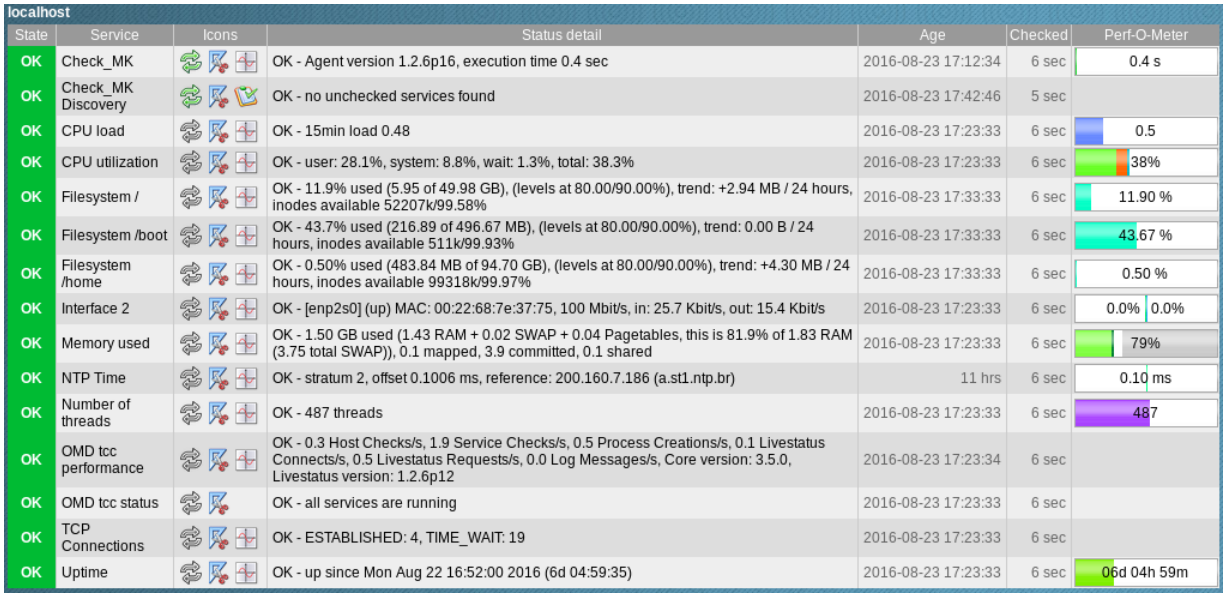


Figura 23 - OMD: Agentes próprios
Fonte: Autoria própria

Além dos agentes para diversas distribuições Unix, o Check_MK também dispõe de agentes para o sistema operacional Windows. Para este trabalho foi efetuado o *download* do pacote com extensão ".rpm", referente ao sistema operacional da estação de gerência. Após efetuar o download do pacote, a instalação foi feita através do comando contido no Apêndice E.

Com a instalação concluída, a estação de gerência foi adicionada aos *hosts* gerenciados. O resultado dos serviços gerenciados através do agente pode ser visto na Figura 24.



localhost

State	Service	Icons	Status detail	Age	Checked	Perf-O-Meter
OK	Check_MK		OK - Agent version 1.2.6p16, execution time 0.4 sec	2016-08-23 17:12:34	6 sec	0.4 s
OK	Check_MK Discovery		OK - no unchecked services found	2016-08-23 17:42:46	5 sec	
OK	CPU load		OK - 15min load 0.48	2016-08-23 17:23:33	6 sec	0.5
OK	CPU utilization		OK - user: 28.1%, system: 8.8%, wait: 1.3%, total: 38.3%	2016-08-23 17:23:33	6 sec	38%
OK	Filesystem /		OK - 11.9% used (5.95 of 49.98 GB), (levels at 80.00/90.00%), trend: +2.94 MB / 24 hours, inodes available 52207k/99.58%	2016-08-23 17:33:33	6 sec	11.90 %
OK	Filesystem /boot		OK - 43.7% used (216.89 of 496.67 MB), (levels at 80.00/90.00%), trend: 0.00 B / 24 hours, inodes available 511k/99.93%	2016-08-23 17:33:33	6 sec	43.67 %
OK	Filesystem /home		OK - 0.50% used (483.84 MB of 94.70 GB), (levels at 80.00/90.00%), trend: +4.30 MB / 24 hours, inodes available 99318k/99.97%	2016-08-23 17:33:33	6 sec	0.50 %
OK	Interface 2		OK - [enp2s0] (up) MAC: 00:22:68:7e:37:75, 100 Mbit/s, in: 25.7 Kbit/s, out: 15.4 Kbit/s	2016-08-23 17:23:33	6 sec	0.0% 0.0%
OK	Memory used		OK - 1.50 GB used (1.43 RAM + 0.02 SWAP + 0.04 Pagetables, this is 81.9% of 1.83 RAM (3.75 total SWAP)), 0.1 mapped, 3.9 committed, 0.1 shared	2016-08-23 17:23:33	6 sec	79%
OK	NTP Time		OK - stratum 2, offset 0.1006 ms, reference: 200.160.7.186 (a.st1.ntp.br)	11 hrs	6 sec	0.10 ms
OK	Number of threads		OK - 487 threads	2016-08-23 17:23:33	6 sec	487
OK	OMD tcc performance		OK - 0.3 Host Checks/s, 1.9 Service Checks/s, 0.5 Process Creations/s, 0.1 Livestatus Connects/s, 0.5 Livestatus Requests/s, 0.0 Log Messages/s, Core version: 3.5.0, Livestatus version: 1.2.6p12	2016-08-23 17:23:34	6 sec	
OK	OMD tcc status		OK - all services are running	2016-08-23 17:23:33	6 sec	
OK	TCP Connections		OK - ESTABLISHED: 4, TIME_WAIT: 19	2016-08-23 17:23:33	6 sec	
OK	Uptime		OK - up since Mon Aug 22 16:52:00 2016 (6d 04:59:35)	2016-08-23 17:23:33	6 sec	06d 04h 59m

Figura 24 - OMD: Serviços monitorados na estação de gerência
Fonte: Autoria própria

Pela Figura 24, percebe-se que o agente do Check_MK permite o gerenciamento de diversos itens do equipamento em que está instalado o agente, aqui, destacam-se os dados referentes ao consumo de recursos do CPU, o uso de memória, o tráfego da placa de rede e também as informações que se referem ao disco.

Pelos fatos apresentados nesta subSeção, atribui-se ao OMD o resultado **sim**, deixando-o na primeira colocação juntamente com o Zabbix.

5.2.3.2 Cacti

Para efetuar o monitoramento de máquinas que não possuam o agente SNMP de forma nativa, o Cacti utiliza uma ferramenta chamada Net-SNMP. O Net-SNMP é uma ferramenta *open-source* disponível tanto para ambientes UNIX, quanto para ambientes *Windows* e que implementa as três versões do protocolo SNMP para realizar o monitoramento dos dispositivos.

O Net-SNMP foi instalado na estação de gerência para monitorá-la através do Cacti; a instalação deste pacote foi realizada previamente ao Cacti, como

apresentado no Apêndice G. Com o Net-SNMP instalado, foi possível monitorar alguns recursos da estação de gerência, como o consumo de memória RAM, o número de usuários conectados no sistema e a quantidade de processos executados. A Figura 25 exibe os gráficos com essas informações, gerados pelo Cacti.

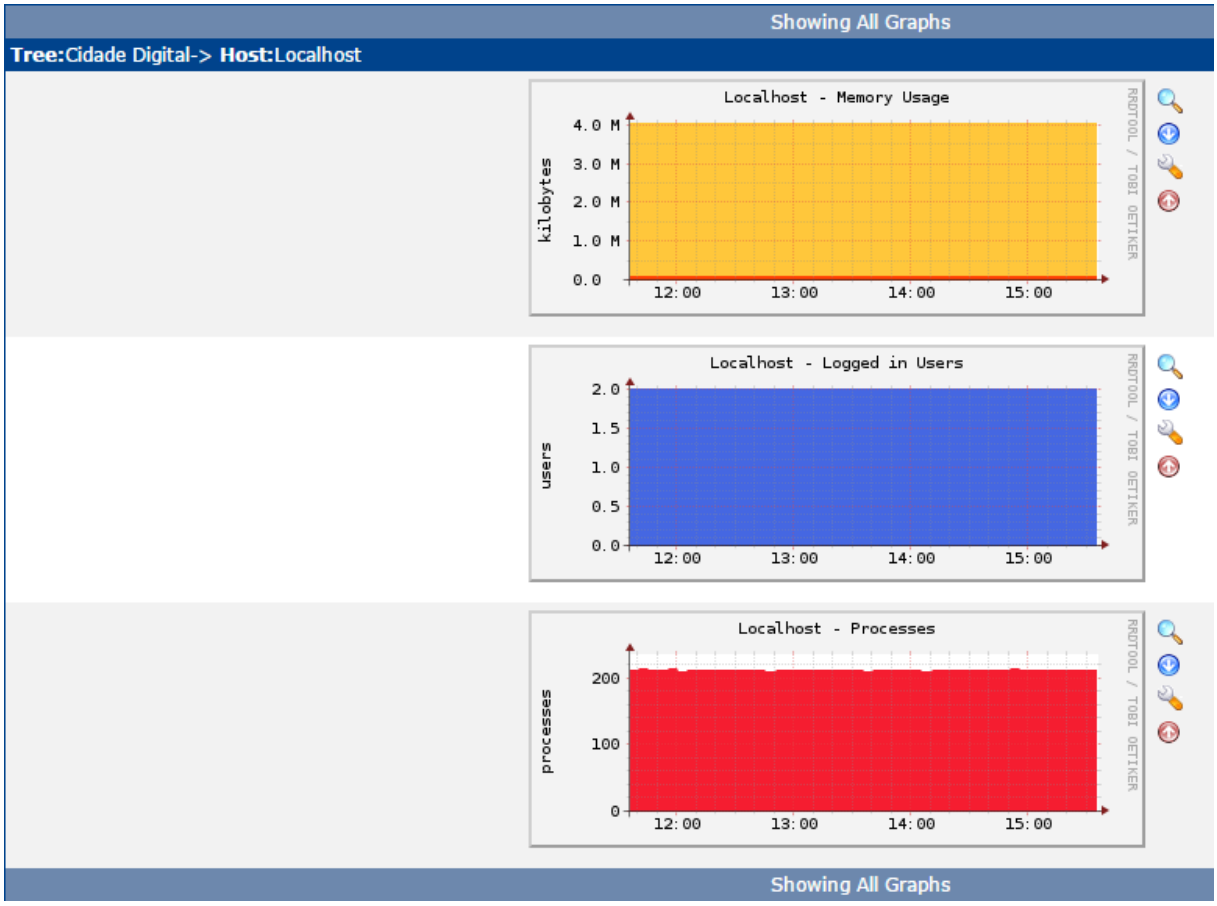


Figura 25 - Cacti: Serviços monitorados na estação de gerência
Fonte: Autoria própria

Como descrito anteriormente e apresentado na Figura 25, os serviços monitorados pelo Cacti através do Net-SNMP são escassos em comparação a outros agentes, como, por exemplo, o agente utilizado pelo OMD. Para efeito comparativo, o agente do OMD monitorou treze serviços, enquanto o Net-SNMP encontrou apenas três. Pelo fato de o Net-SNMP ser um pacote independente do Cacti (que não possui agente próprio), foi atribuído **não** para o valor da métrica estudada. Com este resultado, o Cacti ocupou a segunda colocação neste quesito.

5.2.3.3. Zabbix

Assim como o OMD e diferentemente do Cacti, o Zabbix possui um agente próprio para monitorar dispositivos que não possuam o protocolo SNMP de forma nativa, neste caso, a estação de gerência, o que atribuiu à ferramenta o valor **sim** para a métrica em questão. A instalação do agente Zabbix é simples e pode ser feita diretamente no terminal, buscando do repositório o pacote “zabbix-agent”. Para esta análise, a instalação do agente foi feita durante a instalação da própria ferramenta, com o comando contido no Apêndice O.

Ao concluir a instalação do agente, o Zabbix já é capaz de monitorar o dispositivo. Para isso, basta adicionar um novo *host* com os dados do dispositivo. Após a instalação do agente e do vínculo dos *templates* “Zabbix Server” e “Linux OS”, o Zabbix encontrou 11 aplicações para monitorar (CPU, sistema de arquivos, memória, desempenho, entre outros), 83 itens, 47 *triggers*, gerando um total de 16 gráficos. Para fins de apresentação, a Figura 26 exibe um dos gráficos gerados a partir do agente Zabbix, o qual tratou do uso da CPU.

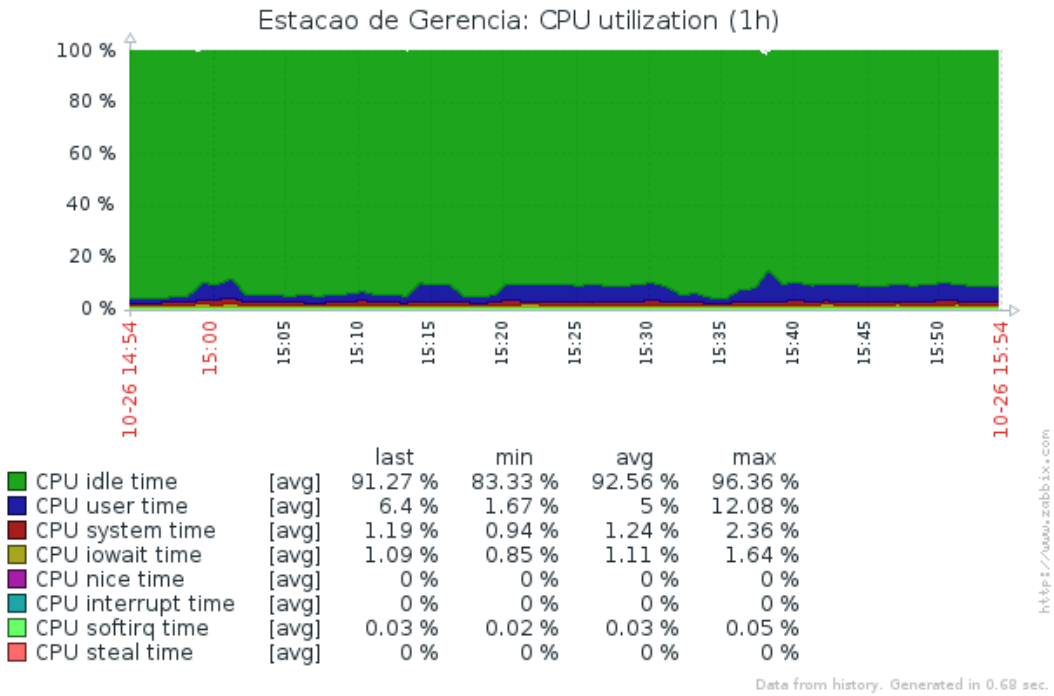


Figura 26 - Zabbix: Utilização de CPU na estação de gerência
Fonte: Autoria própria

Pelo fato de assim como o OMD, o Zabbix possuir agente próprio, as duas ferramentas empataram na primeira colocação.

O Quadro 9 resume os resultados obtidos na avaliação desta métrica.

Ferramenta	Agente próprio	Posição da Ferramenta
OMD	Sim	1ª colocação
Zabbix	Sim	1ª colocação
Cacti	Não	2ª colocação

Quadro 9 - Resultado: Agente próprio
Fonte: Autoria própria

5.2.4 Grau de dificuldade

Apresenta os resultados obtidos das avaliações sob as características expostas no item de mesmo nome, contido na Seção 4.2.2.2.

5.2.4.1 OMD

Para adicionar os *hosts* que foram monitorados pelo OMD durante o trabalho não foi necessária nenhuma pesquisa, já que essa opção se apresenta de forma explícita no menu principal WATO. Ao acessar o WATO, a segunda opção disponível ao administrador é o ícone dos *hosts*. Acessando essa página, o administrador tem à sua disposição várias opções relativas aos *hosts*.

Para adicionar um novo *host,* o administrador deve selecionar a opção "*New host*", com isso, o administrador será levado a uma nova página e deve inserir os dados do *host* a ser monitorado. Nesta página há alguns dados que o administrador deve inserir para adicionar o *host*, sendo eles:

- *Hostname*: nome que o administrador quer atribuir ao *host*. Esse é o nome que deve ser inserido nas regras de monitoramento, bem como em outras opções de configurações do OMD;
- *Permissions*: atribui permissões ao *host* para definir as configurações de notificações;
- *Alias*: "apelido" dado ao *host* utilizado somente para visualização de gráficos e outras informações na ferramenta;
- *IP address*: endereço IP do dispositivo a ser monitorado;

- *Parents*: o Check_MK permite que o administrador atribua “pais” ao *host*. Um “pai” é um dispositivo que está hierarquicamente acima de determinado *host* na topologia da rede em questão. Neste trabalho, foi determinado que o *switch* é “pai” de todas estações *wireless*;
- *Agent type*: o administrador deve selecionar com qual agente aquele *host* será monitorado. Por padrão, o Check_MK atribui seu próprio agente, porém, o único dispositivo gerenciado através do agente foi a própria estação de gerência, todos os outros dispositivos foram gerenciados utilizando o agente SNMPv2. Caso o dispositivo não aceite nenhuma das opções citadas, o recurso utilizado para monitoramento será o *ping*;
- *Criticality*: o Check_MK permite que o administrador vincule o dispositivo a um tipo de sistema, com o objetivo de personalizar o tempo das notificações emitidas. Para o desenvolvimento do trabalho foi utilizado o valor padrão de “*Productive system*” para todos os dispositivos;
- *Networking Segment*: também há a opção para que o administrador defina a qual segmento de rede aquele dispositivo que está sendo configurado pertence. Todos os dispositivos gerenciados durante o trabalho receberam o segmento “*Local network*”.

A Figura 27 apresenta a página para a adição de um novo *host*.

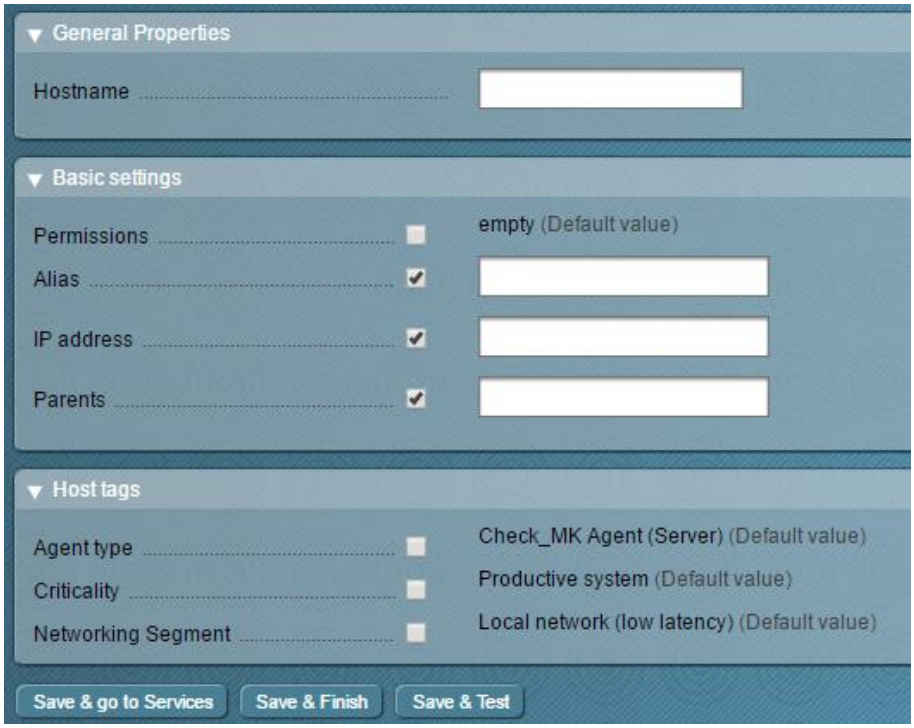


Figura 27 - OMD: Novo *host*
Fonte: Autoria própria

Ao término da inserção dessas configurações, o Check_MK já está hábil a obter os serviços disponíveis no agente. Segundo a documentação oficial do Check_MK, em dispositivos onde é utilizado o agente próprio da ferramenta, o gerente simplesmente obtém a saída do agente e retorna ao usuário os serviços disponíveis. Já em dispositivos onde é utilizado o agente SNMP, o processo é um pouco diferente.

A checagem dos serviços disponíveis através do agente SNMP é feita em duas etapas: a primeira é a consulta apenas dos dois primeiros identificadores de objetos (*sysDescr* e *sysObjectID*) contidos na MIB do dispositivo e, dependendo desses dados, busca aproximadamente mais dez objetos. Baseado nos resultados obtidos é que o gerente irá verificar quais dos aproximadamente 600 plug-ins SNMP padrões são suportados pelo dispositivo gerenciado. A segunda etapa feita pelo gerente é a coleta de todos os dados necessários através de comandos "*SNMP-walks*", são esses dados que determinam quais serão os serviços monitorados.

Nos dispositivos gerenciados durante o desenvolvimento deste trabalho, a descoberta dos serviços disponíveis durou poucos segundos, por se tratarem de dispositivos com poucas interfaces e também por estarem em um ambiente de rede local. Na estação de gerência foram monitorados os seguintes serviços: carga de CPU, uso de CPU, tráfego na placa de rede, consumo de memória, número de *threads*, desempenho do OMD, status do OMD, quantidade de conexões TCP e o *uptime*. Nas estações de rádio foram monitorados: o tráfego da interface *ethernet*, o tráfego da interface *wireless*, as informações do SNMP e o *uptime*. No *switch*, foram monitoradas todas as interfaces F*astEthernet* e G*igabitEthernet*, o uso de CPU, o consumo de memória, o consumo de memória do CPU, as informações do SNMP e o *uptime*.

Todos os dispositivos gerenciados durante o trabalho tiveram o mesmo processo para configuração na ferramenta, por isso e pelos fatos apresentados nesta Seção, foi concedido à ferramenta o valor de **grau I** na referida métrica, colocando-a empatada com as outras ferramentas, na primeira colocação neste critério.

5.2.4.2 Cacti

Assim como o OMD, o Cacti também exibe de forma explícita a opção para a adição dos *hosts* a serem monitorados. Logo na página inicial, ao término da instalação da ferramenta, há três *links* com atalhos para algumas ações básicas, como a criação dos dispositivos, criação de gráficos e a visualização dos gráficos gerados. Ao clicar para adicionar um novo *host*, algumas informações a respeito do dispositivo são exigidas pela ferramenta, como mostra a Figura 28.

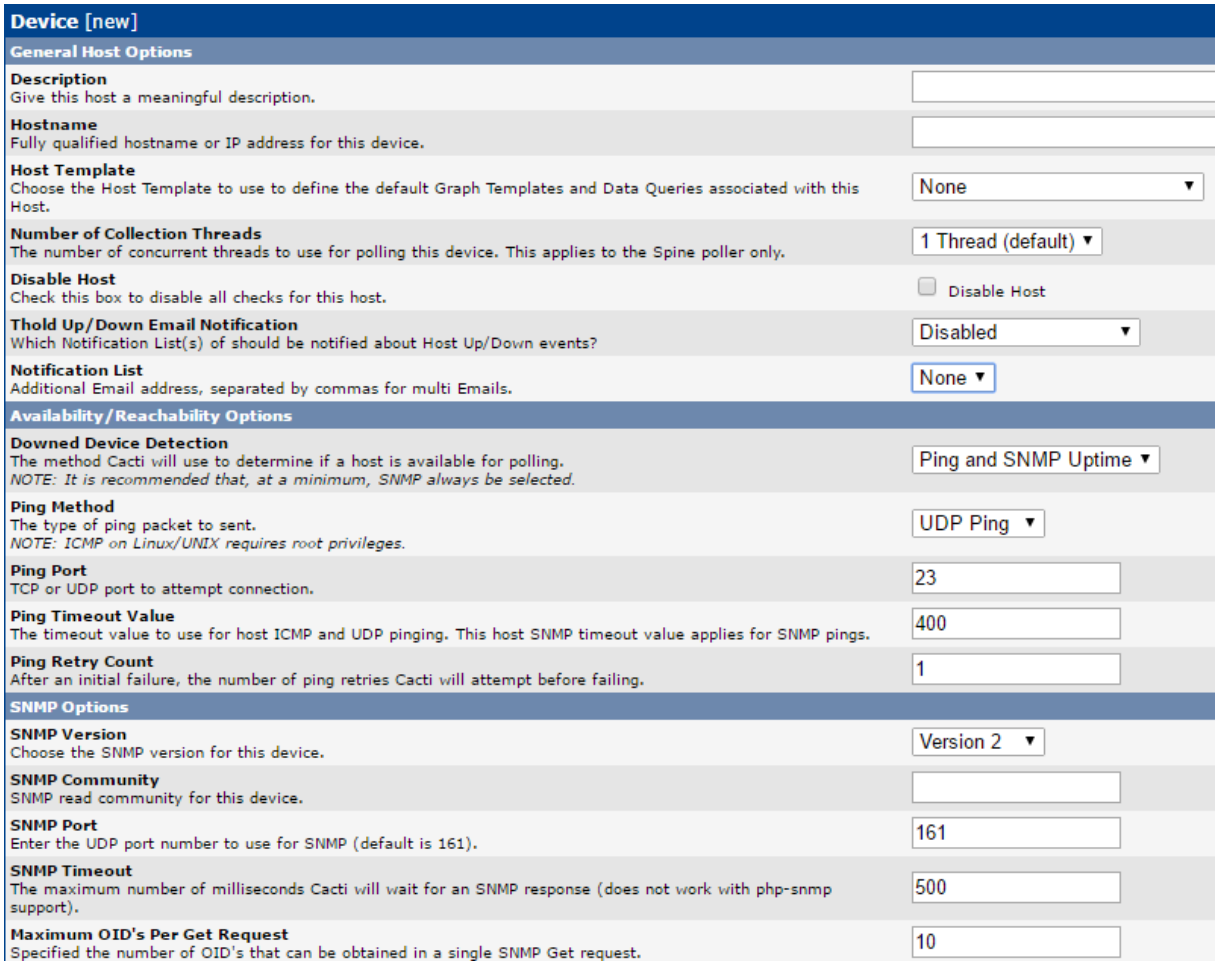


Figura 28 - Cacti: Novo *host*
Fonte: Autoria própria

As informações a respeito do dispositivo se dividem em informações gerais a respeito do *host*, informações para avaliar o status do *host* (*up* ou *down*) e opções referentes ao protocolo SNMP. Segue abaixo a relação das opções que receberam alterações, as opções não citadas abaixo e que são apresentadas na Figura 28 ficaram com seus valores padrão.

- *Description*: nome atribuído ao *host* gerenciado. Este nome é utilizado para a exibição do dispositivo no Cacti;
- *Hostname*: endereço IP ou *hostname* do dispositivo gerenciado. Neste trabalho todos os dispositivos foram adicionados pelo endereço IP;
- *Host template*: os gráficos gerados pelo Cacti são baseados em *templates* (modelos), essa opção permite que o administrador vincule o dispositivo a *templates* padrões do Cacti. Todas as estações de rádio receberam o *template* "*Generic SNMP-enabled Host*", enquanto a estação de gerência recebeu o "*Local Linux Machine*" e o *switch* foi atrelado ao *template* "*Cisco Router*";
- *Downed Device Detection*: método utilizado pelo Cacti para determinar se o *host* está hábil a receber o *poller*. Todos os dispositivos gerenciados foram configurados para utilizar o *ping* e o SNMP U*ptime* para que isso seja determinado;
- SNMP *Version*: deve ser configurada qual versão do SNMP o dispositivo utiliza. Todos os dispositivos gerenciados durante o trabalho utilizaram o SNMPv2;
- SNMP *Community*: se houver, deve ser inserida a comunidade do SNMP. Por questões de segurança, a comunidade utilizada pelos dispositivos foi ocultada do trabalho.

Ao término dessa configuração, o Cacti já está hábil a receber os dados do dispositivo, armazenar essas informações e gerar os gráficos. Para obter os dados do dispositivo, por padrão, o Cacti executa *SNMPQueries* que retornam todos os identificadores de objeto disponíveis no dispositivo. A definição das *SNMPQueries* que serão executadas para determinado dispositivo é feita através do *template* associado ao mesmo. Além dos *SNMPQueries*, o Cacti é capaz de executar *scripts* externos escritos em diversas linguagens para obter as informações dos dispositivos, no entanto, este recurso não foi utilizado ao longo do desenvolvimento desta análise.

A Figura 29 apresenta os valores obtidos do objeto .1.3.6.1.2.1.2.2.1.2 (*ifDescr*) retornados após a configuração do *switch*. Esses valores ficam armazenados pelo Cacti e podem ser utilizados dentro da ferramenta, neste caso, os valores obtidos do objeto a seguir foram utilizados no título dos gráficos gerados.

```
+ Executing SNMP walk for data @ '.1.3.6.1.2.1.2.2.1.2'
+ Found item [ifDescr='Vlan1'] index: 1 [from value]
+ Found item [ifDescr='Vlan2'] index: 2 [from value]
+ Found item [ifDescr='FastEthernet0/1'] index: 10001 [from value]
+ Found item [ifDescr='FastEthernet0/2'] index: 10002 [from value]
+ Found item [ifDescr='FastEthernet0/3'] index: 10003 [from value]
+ Found item [ifDescr='FastEthernet0/4'] index: 10004 [from value]
+ Found item [ifDescr='FastEthernet0/5'] index: 10005 [from value]
+ Found item [ifDescr='FastEthernet0/6'] index: 10006 [from value]
+ Found item [ifDescr='FastEthernet0/7'] index: 10007 [from value]
+ Found item [ifDescr='FastEthernet0/8'] index: 10008 [from value]
+ Found item [ifDescr='FastEthernet0/9'] index: 10009 [from value]
+ Found item [ifDescr='FastEthernet0/10'] index: 10010 [from value]
+ Found item [ifDescr='FastEthernet0/11'] index: 10011 [from value]
+ Found item [ifDescr='FastEthernet0/12'] index: 10012 [from value]
+ Found item [ifDescr='FastEthernet0/13'] index: 10013 [from value]
+ Found item [ifDescr='FastEthernet0/14'] index: 10014 [from value]
+ Found item [ifDescr='FastEthernet0/15'] index: 10015 [from value]
+ Found item [ifDescr='FastEthernet0/16'] index: 10016 [from value]
+ Found item [ifDescr='FastEthernet0/17'] index: 10017 [from value]
+ Found item [ifDescr='FastEthernet0/18'] index: 10018 [from value]
+ Found item [ifDescr='FastEthernet0/19'] index: 10019 [from value]
+ Found item [ifDescr='FastEthernet0/20'] index: 10020 [from value]
+ Found item [ifDescr='FastEthernet0/21'] index: 10021 [from value]
+ Found item [ifDescr='FastEthernet0/22'] index: 10022 [from value]
+ Found item [ifDescr='FastEthernet0/23'] index: 10023 [from value]
+ Found item [ifDescr='FastEthernet0/24'] index: 10024 [from value]
+ Found item [ifDescr='GigabitEthernet0/1'] index: 10101 [from value]
+ Found item [ifDescr='GigabitEthernet0/2'] index: 10102 [from value]
+ Found item [ifDescr='Null0'] index: 10501 [from value]
```

Figura 29 - Cacti: Exemplo de objetos retornados por um SNMPQuery
Fonte: Autoria própria

Destaca-se que durante esta análise que os únicos dispositivos monitorados via SNMP foram o *switch* e a estação de gerência, não sendo possível monitorar as estações de rádio através do protocolo. Na estação de gerência foram monitorados os serviços: consumo de memória, quantidade de usuários conectados e número de processos. Nas estações de rádio o único serviço monitorado foi o tempo de resposta através do utilitário Ping. No *switch*, foram monitoradas todas as interfaces F*astEthernet* e G*igabitEthernet*, as informações do SNMP e o *uptime*.

Pelos fatos apresentados e respeitando os critérios estabelecidos na Seção 4.2.2.1, o Cacti recebeu **grau I** para a dificuldade com a configuração dos *hosts* e serviços, posicionando-se também na primeira colocação.

5.2.4.3 Zabbix

Desconsiderando a opção de adição dos *hosts* de forma automática com uma *action* e a opção de autodescoberta, a adição de forma manual de um *host* na ferramenta Zabbix é feita através do menu “*Configuration*” > “*Hosts*”. Ao acessar essa página, é apresentado ao usuário todos os *hosts* que já estão sendo monitorados e também uma opção “*Create new host*”; clicando nessa opção, o usuário é levado à página para o cadastro de um novo *host* a ser monitorado.

A inclusão de um novo *host* se divide em seis categorias, sendo elas:

- *Host*: nessa primeira etapa, as informações inseridas dizem respeito diretamente ao *host* a ser cadastrado. Foram inseridos os valores para o *hostname*, o *alias*, o *hostgroups* ao qual o novo *host* foi vinculado e o endereço IP do *host*;
- *Templates*: devem ser inseridos os *templates* que serão vinculados ao *host*. Para o desenvolvimento deste trabalho, todos os *hosts* foram vinculados ao *template* ICMP Ping, sendo que o *switch* recebeu um *template* específico obtido da comunidade Zabbix e a estação de gerência recebeu um *template* para servidores Zabbix e outro para máquinas Linux;
- IPMI (*Intelligent Platform Management Interface*): segundo a própria documentação do Zabbix, o IPMI é uma interface independente do servidor que permite o monitoramento do hardware através dos cartões de gerenciamento, sem necessidade de interferência do sistema operacional. Para esta análise, este recurso não foi utilizado considerando-se que os dispositivos utilizados no cenário não possuem suporte a este recurso.
- Macros: funcionam como “variáveis globais” dentro do Zabbix. Após a definição de um macro ({$MACRO}) com seu respectivo valor, esse macro pode ser utilizado em *templates*, gráficos, *actions* ou qualquer outra configuração que as aceite. Novamente, nenhum macro foi personalizado durante esta análise;
- *Host inventory*: o Zabbix tem um grande recurso para documentação da rede, sendo um desses recursos o “*host inventory*”, neste item o administrador da rede pode inserir todas as informações acerca do

dispositivo, como tipo, modelo, números de série, endereço MAC (*Media Access Control*), entre diversas outras informações. O *"host inventory"* é um grande aliado ao administrador no que compete ao Gerenciamento de Configuração. Sendo que o Gerenciamento de Configuração não é o foco desta análise, esse recurso não foi utilizado;

- *Encryption*: se o dispositivo a ser adicionado possui alguma criptografia que restrinja seu monitoramento, deve ser configurado neste item.

Após a inserção de todas as informações acerca do *host*, basta clicar em "*Add"* para que o novo *host* seja adicionado e passe a ser monitorado pelo Zabbix.

Para a obtenção dos dados, o Zabbix utiliza os itens definidos pelo *template*. Por exemplo, para monitorar a entrada de dados na interface *Gibabit* do *switch*, o *template* faz uso do identificador de objeto .1.3.6.1.2.1.2.2.1.10.10101 (*ifInOctets* da interface 10101), utilizando o protocolo SNMPv2. Caso seja necessário adicionar alguma checagem que não há no *template*, basta o administrador criar um novo item inserindo as informações de acordo com a MIB do dispositivo. Além do uso do SNMP, o Zabbix pode fazer checagens simples, utilizando, por exemplo, o protocolo ICMP para realizar checagens através do Ping.

Assim como o Cacti, o Zabbix também não foi capaz de monitorar via SNMP as estações de rádio, realizando o monitoramento destas apenas pelo Ping. Enquanto a estação de gerência e o *switch* foram monitorados, respectivamente, pelo agente Zabbix e o agente nativo do protocolo SNMPv2. Na estação de gerência foram monitorados os serviços de: carga de CPU, utilização de CPU, consumo de memória, tráfego na placa de rede, uso de *swap* e o desempenho do próprio Zabbix. No *switch* foram monitoradas todas as interfaces *FastEthernet* e *GigabitEthernet*, o tempo de resposta através do Ping e o *uptime*.

Por todos os fatos expostos durante esta análise, atribui-se à ferramenta Zabbix o **grau I** no que compete a dificuldade para a configuração dos *hosts*, mantendo as três ferramentas analisadas empatadas na primeira colocação.

O Quadro 10 exibe o resultado das avaliações da referida métrica.

Ferramenta	**Grau de dificuldade na configuração dos *hosts***	**Posição da Ferramenta**
OMD	Grau I	1ª colocação
Cacti	Grau I	1ª colocação
Zabbix	Grau I	1ª colocação

Quadro 10 - Resultado: Grau de dificuldade na configuração dos *hosts*
Fonte: Autoria própria

5.3 GERAÇÃO DE INFORMAÇÕES

Esta Seção apresenta as avaliações e resultados referentes ao critério de geração de informações, respeitando as métricas previamente estabelecidas na Seção 4.2.2.3.

5.3.1 Mapas e/ou diagramas de rede

Expõe-se nesta Seção, as avaliações e resultados obtidos da análise das três ferramentas, respeitando os critérios estabelecidos na definição da referida métrica realizada no capítulo anterior.

5.3.1.1 OMD

Por padrão, a geração de mapas e diagramas no OMD é responsabilidade de uma ferramenta chamada NagVis. De acordo com a própria documentação, o NagVis é uma ferramenta gratuita, capaz de gerar diagramas e mapas com os serviços e dispositivos monitorados pelo Nagios. Para efetuar a análise desta métrica, foi realizada a tentativa de criação de um diagrama de rede automático e também a criação de um mapa personalizado, de acordo com as características desejadas pelo autor.

Os mapas configurados no NagVis têm seus títulos exibidos na barra lateral de navegação do Check_MK e, por padrão, o NagVis já traz configurados seis mapas para exemplo. Além dos mapas padrão, o NagVis automaticamente gera diagramas da topologia das redes monitoradas pelo OMD, separando-as pelas pastas em que os *hosts* estão agrupados. Pelo fato de haver apenas uma rede gerenciada no desenvolvimento do trabalho, todos os dispositivos estão agrupados na pasta principal. Os diagramas gerados podem ser acessados através do menu “*dashboards*”, no *link* “*Network Topology*”.

O diagrama gerado durante o trabalho pode ser visualizado na Figura 30.

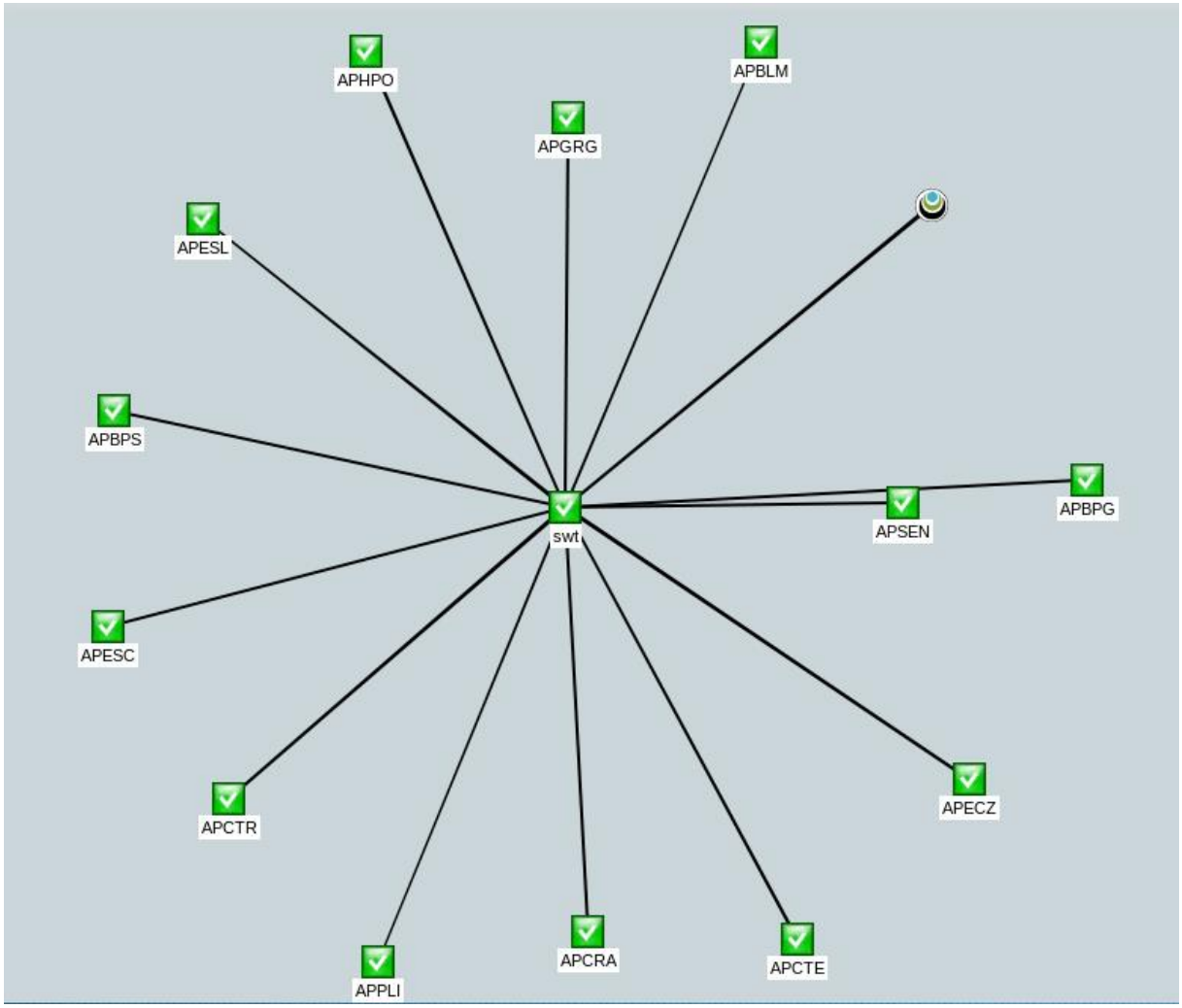
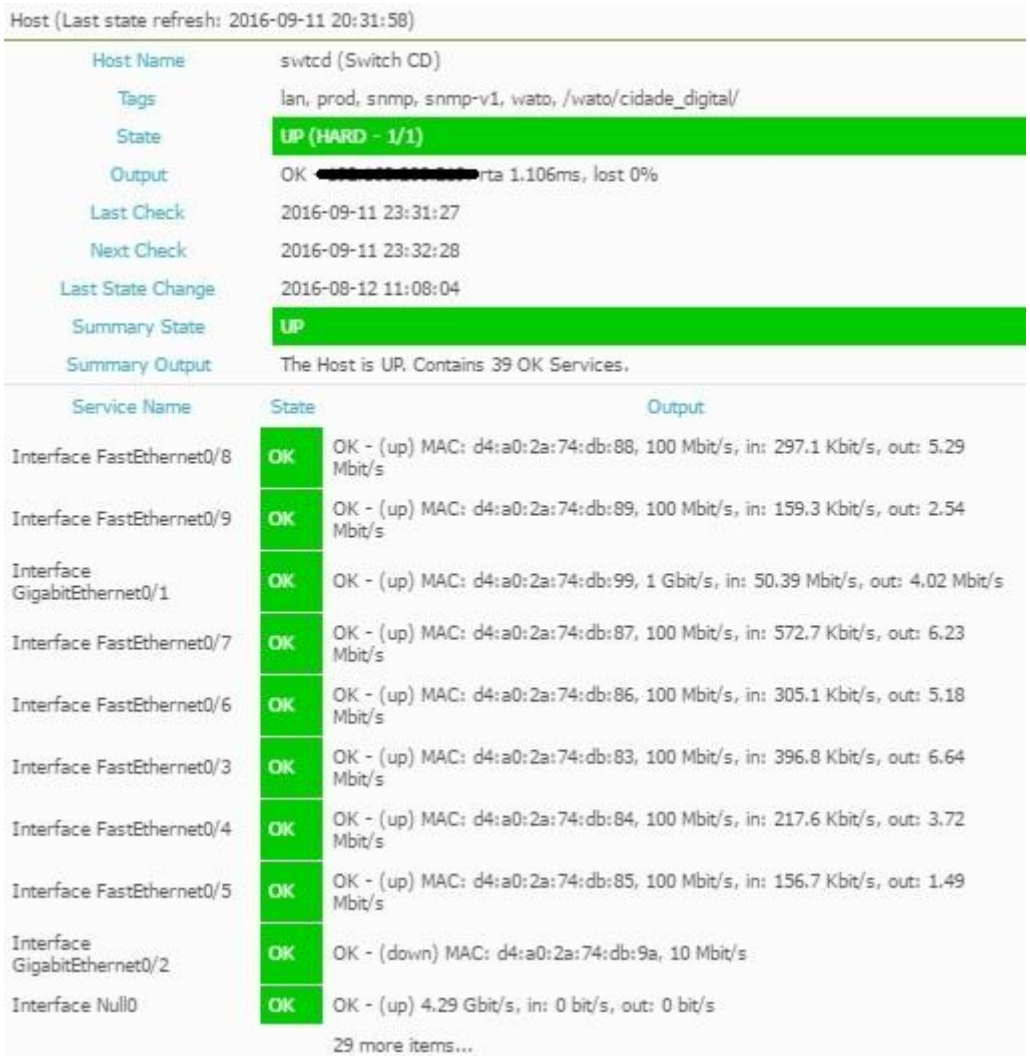


Figura 30 - OMD: Diagrama automático de rede
Fonte: Autoria própria

Pelo fato de que todos os *hosts* inseridos tiveram o *switch* como “pai”, o NagVis cria o diagrama com o *switch* centralizado e todos os outros dispositivos ligados a ele. Além dos dispositivos gerenciados, também é exibido o ícone do NagVis que representa a estação de gerência. O diagrama gerado apresenta um ícone para cada *host*, sendo que sua cor representa o status do mesmo, sendo que o ícone verde representa o status *up* e o vermelho, o status *down*.

Além de visualizar o status do *host* diretamente no diagrama, quando deixado o mouse sob um dos ícones, o NagVis apresenta um pequeno pop-up com um resumo do *host*. Este resumo apresenta informações a respeito dos serviços gerenciados naquele *host*, sobre o horário das checagens, o status do *host*, entre outras informações.

A Figura 31 apresenta o *pop-up* com as informações sobre o *switch*, sendo que por motivos de segurança, o endereço IP foi ocultado.



Host (Last state refresh: 2016-09-11 20:31:58)

Host Name	swtcd (Switch CD)
Tags	lan, prod, snmp, snmp-v1, wato, /wato/cidade_digital/
State	UP (HARD - 1/1)
Output	OK [redacted]rta 1.106ms, lost 0%
Last Check	2016-09-11 23:31:27
Next Check	2016-09-11 23:32:28
Last State Change	2016-08-12 11:08:04
Summary State	UP
Summary Output	The Host is UP. Contains 39 OK Services.

Service Name	State	Output
Interface FastEthernet0/8	OK	OK - (up) MAC: d4:a0:2a:74:db:88, 100 Mbit/s, in: 297.1 Kbit/s, out: 5.29 Mbit/s
Interface FastEthernet0/9	OK	OK - (up) MAC: d4:a0:2a:74:db:89, 100 Mbit/s, in: 159.3 Kbit/s, out: 2.54 Mbit/s
Interface GigabitEthernet0/1	OK	OK - (up) MAC: d4:a0:2a:74:db:99, 1 Gbit/s, in: 50.39 Mbit/s, out: 4.02 Mbit/s
Interface FastEthernet0/7	OK	OK - (up) MAC: d4:a0:2a:74:db:87, 100 Mbit/s, in: 572.7 Kbit/s, out: 6.23 Mbit/s
Interface FastEthernet0/6	OK	OK - (up) MAC: d4:a0:2a:74:db:86, 100 Mbit/s, in: 305.1 Kbit/s, out: 5.18 Mbit/s
Interface FastEthernet0/3	OK	OK - (up) MAC: d4:a0:2a:74:db:83, 100 Mbit/s, in: 396.8 Kbit/s, out: 6.64 Mbit/s
Interface FastEthernet0/4	OK	OK - (up) MAC: d4:a0:2a:74:db:84, 100 Mbit/s, in: 217.6 Kbit/s, out: 3.72 Mbit/s
Interface FastEthernet0/5	OK	OK - (up) MAC: d4:a0:2a:74:db:85, 100 Mbit/s, in: 156.7 Kbit/s, out: 1.49 Mbit/s
Interface GigabitEthernet0/2	OK	OK - (down) MAC: d4:a0:2a:74:db:9a, 10 Mbit/s
Interface Null0	OK	OK - (up) 4.29 Gbit/s, in: 0 bit/s, out: 0 bit/s
		29 more items...

Figura 31 - OMD: Pop-up com resumo do host
Fonte: Autoria própria

Além do diagrama de rede automático, também foi avaliada a capacidade da ferramenta quanto a criação de mapas personalizáveis. O objetivo final era obter um mapa com a distribuição geográfica das estações *wireless* pela cidade, além de exibir, em tempo real, o tráfego de rede de cada uma das estações, através do monitoramento das interfaces do *switch*. Para elaborar este tipo de mapa, o NagVis conta com um recurso para adicionar *background*, neste caso, uma imagem de satélite da cidade. Os passos utilizados para a criação deste mapa estão descritos a seguir.

- No menu “*options*”, em “*Manage backgrounds*”, foi efetuado o *upload* da imagem que serviu de *background* ao mapa;

- Em “*Manage maps*” foi criado um novo mapa, inserido um ID, um nome e selecionado seu tipo, nessa situação, um “*Regular map*”;

Com esses passos, o mapa já estava criado, porém, não possuía nenhuma informação. Os passos descritos a seguir remetem à configuração do mapa criado.

- A inserção do *background* foi realizada através do menu principal, na opção “*Edit map*” e, em seguida, navegou-se ao menu “*Map options*”. Em “*map_image*” é apresentado um *combo box* com as imagens que estão disponíveis no servidor. Selecionando a imagem e salvando as novas configurações, o mapa já apresentava o *background*;

Com o *background* inserido no mapa, foram adicionados os *hosts* monitorados. Essa etapa foi executada para todos os *hosts*.

- No menu “*Edit map*”, em “*Add icon*”, “*host*”. Ao clicar em *host*, o NagVis gera um cursor na tela que deve ser utilizado para adicionar o *host* ao mapa. Clicando em uma posição, o NagVis apresenta o menu de configuração do *host*. A primeira opção é um *combo box* com a lista de todos os *hosts* que estão sendo gerenciados pelo OMD; selecionou-se apenas o *host_name*. Após salvar, o *host* já pôde ser visualizado no mapa.

Para monitorar o *throughput* em tempo real de cada estação de rádio, foram adicionadas linhas que remetem a cada interface do *switch*. Esse processo é parecido com a configuração dos *hosts*, entretanto, é utilizado o monitoramento do serviço de um *host*, não somente um *host*.

Tanto os ícones inseridos para os *hosts*, quanto as linhas que exibem os serviços, seguem o mesmo padrão de exibição apresentado no diagrama automático, como cores para o status do elemento gerenciado e *pop-ups* com um resumo do mesmo.

O resultado da criação deste mapa é exibido na Figura 32.

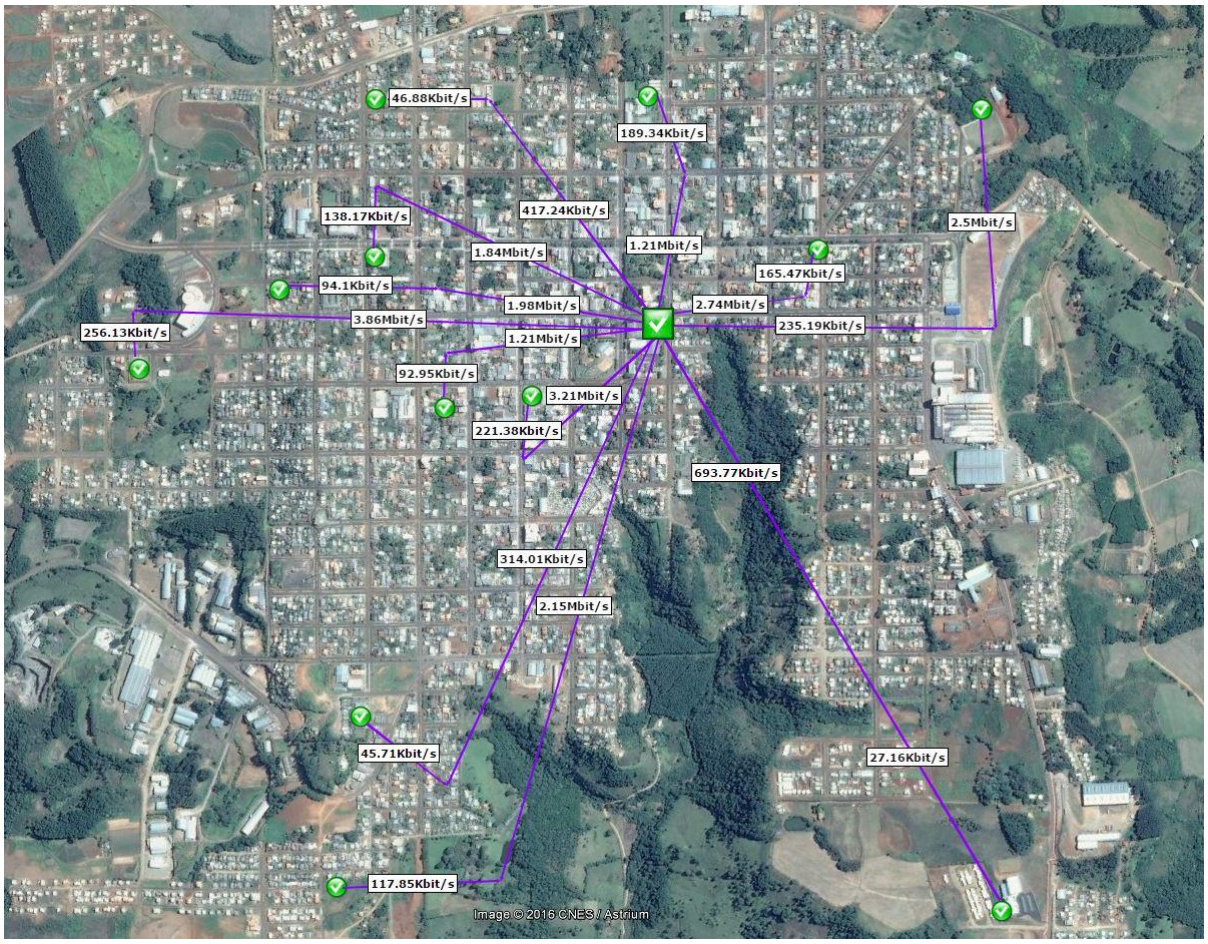


Figura 32 – OMD: Mapa com a distribuição geográfica dos *hosts*
Fonte: Autoria própria

Por todos os fatos apresentados, o OMD foi classificado como uma **boa** ferramenta no que diz respeito à criação de mapas e diagramas de rede, já que possui um diagrama automático dos equipamentos gerenciados e também permite a criação de mapas totalmente personalizáveis. Devido ao resultado obtido, o OMD ocupou a primeira colocação na referida métrica. Durante esta análise foram desenvolvidos e apresentados apenas a criação de um mapa com a distribuição geográfica dos *hosts* e um diagrama automático, visto que a métrica estipulada se restringiu a isso, porém, a ferramenta possui outros recursos e outras capacidades, que poderão ser expostas em um trabalho futuro.

5.3.1.2 Cacti

O Cacti não possui de forma nativa a opção para a criação de mapas e diagramas dos dispositivos monitorados. Há diversos *plug-ins* externos capazes de executar essas funções, no entanto como citado anteriormente, este trabalho foi desenvolvido com as ferramentas em suas versões padrão, sem a adição de *plug-ins* externos. Pelo fato de não possuir este recurso, o Cacti recebeu avaliação **ruim** para esta métrica, posicionando-se na terceira colocação.

5.3.1.3 Zabbix

O Zabbix não tem a capacidade de gerar mapas e diagramas de rede de forma automática, mas permite que o administrador crie seus mapas personalizados dentro da ferramenta. A criação dos mapas é feita diretamente pelo Zabbix, diferentemente do OMD, por exemplo, que delega a criação de mapas a outra ferramenta.

Para acessar os mapas criados e também para criar um novo mapa, deve-se acessar o menu “*Monitoring*” > “*Maps*”. Nesta página são exibidos todos os mapas criados, a opção “*Create map*”, para que seja criado um novo mapa, e também a opção “*Import*”, para que o administrador possa importar um mapa já criado. O Zabbix permite, ainda, que qualquer mapa criado possa ser exportado para um arquivo do tipo XML (eXtensible Markup Language).

A criação de mapas no Zabbix é simples e não requer muito conhecimento da ferramenta. As principais informações requeridas são:

- *Owner*: qual usuário Zabbix será o proprietário do mapa;
- *Name*: nome do mapa a ser criado;
- *Width* e *Height*: largura e altura (em pixel) do mapa;
- *Background image*: caso o administrador opte por usar uma imagem de fundo para o mapa, ela deve ser selecionada nesta opção;

Entre outras opções de visualização, que para fins de análise, mantiveram-se com os valores padrões.

Para iniciar a inserção de itens ao mapa, devem ser adicionados os ícones que representam os *hosts* monitorados. Para inserir um novo ícone deve-se

selecionar a opção “*Icon*” > “*Add*”. Uma nova janela abrirá com as opões para o novo ícone a ser adicionado. Em “*Icon type”* deve-se selecionar a opção “*host”* e, em seguida, selecionar qual *host* será representado por aquele ícone. Esse processo deve ser feito para todos os *hosts* que irão ser apresentados no mapa, neste caso, todas as estações de rádio e o *switch* que as gerencia.

Outra opção utilizada são os *links*, que são representações de *triggers* por linhas entre dois ícones. Para adicioná-las, deve-se selecionar dois ícones do mapa e utilizar a opção “*Link”* > *“Add”*, abrirá uma nova janela para o administrador selecionar qual *trigger* aquele *link* representará, no caso, a *trigger* que informa que a estação de rádio está indisponível via ICMP. Essa opção facilita a visualização de eventuais problemas com a rede, o administrador pode, por exemplo, criar uma *trigger* para quando o *throughput* ultrapassar determinado valor pré-estabelecido e utilizá-la como um *link* no mapa, que informará de forma explícita quando aquele *host* estiver com sua banda no limite.

O mapa criado com a distribuição geográfica das estações de rádio, pode ser visualizado na Figura 33.

Figura 33 - Zabbix: Mapa com a distribuição geográfica dos *hosts*
Fonte: Autoria própria

Por não possuir nenhum recurso para geração de diagramas e/ou mapas automáticos dos *hosts*, o Zabbix recebeu avaliação **regular** neste quesito. Vale ressaltar que a criação de apenas um mapa deu-se pelo fato da análise não exigir mais requisitos. Com esta avaliação, o Zabbix ocupou a segunda colocação.

O Quadro 11 exibe o resultado das avaliações da referida métrica.

Ferramenta	Mapas e diagramas de rede	Posição da Ferramenta
OMD	Boa	1ª colocação
Zabbix	Regular	2ª colocação
Cacti	Ruim	3ª colocação

Quadro 11 - Resultado: Mapas e diagramas de rede
Fonte: Autoria própria

5.3.2 Geração de gráficos

Ao utilizar ferramentas de monitoramento e gerenciamento, um dos principais recursos buscados pelos administradores é a geração de gráficos para os dispositivos gerenciados. Esta Seção apresenta as avaliações e resultados obtidos, seguindo os critérios pré-estabelecidos na Seção 4.2.2.3.

5.3.2.1 OMD

A geração de gráficos no OMD é responsabilidade de uma ferramenta *open-source* chamada RRDTool (*Round Robin Database Tool*). “O RRDTool trabalha com uma fila circular, fazendo com que o tamanho da base de dados seja fixo durante todo o seu tempo de vida” (BALBINOT e ANDRADE, 2000). Os gráficos gerados pelo RRDTool são dinâmicos e possuem diversas informações.

Por padrão todos os *hosts* e serviços monitorados pelo OMD já passam a gerar gráficos. Além dos gráficos gerados através do RRDTool, o Check_MK apresenta um pequeno gráfico ao lado de cada serviço, chamado de *Perf-O-Meter*, como mostra a Figura 34.

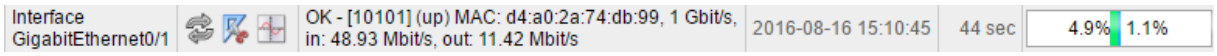


Figura 34 - *Perf-O-Meter* da interface Gigabit do *switch*
Fonte: Autoria própria

Para todos os serviços monitorados que remetem a taxas de transferências são exibidos três gráficos: o primeiro apresenta a taxa de transferência (*download* e *upload*), o segundo apresenta os pacotes e o terceiro, os erros que ocorreram neste serviço. Além disso, os gráficos são categorizados por padrão com seis períodos de tempo: 4 horas, 25 horas, uma semana, um mês e um ano. Porém, qualquer um dos gráficos possui o recurso de zoom, basta o administrador selecionar parte do gráfico para visualizar um período de tempo específico. A Figura 35 apresenta o gráfico com o tráfego da interface gigabit do *switch* durante o período da análise (sete dias).

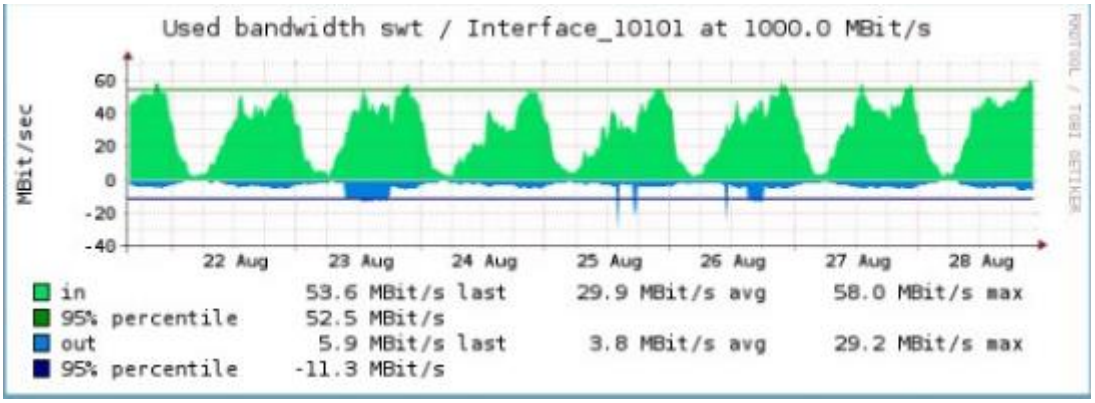


Figura 35 - OMD: Gráfico com o tráfego da interface gigabit do *switch*
Fonte: Autoria própria

Os gráficos gerados pelo OMD seguem as regras estabelecidas aos *hosts* e serviços, portanto, percebe-se que neste caso já seguem a unidade de *bits* como padrão, bem como, já estão estabelecidas as velocidades das interfaces *wireless* das estações de rádio. A Figura 36 apresenta o exemplo com a interface *wireless* de uma das estações monitoradas, com velocidade padrão definida em 54Mbps.

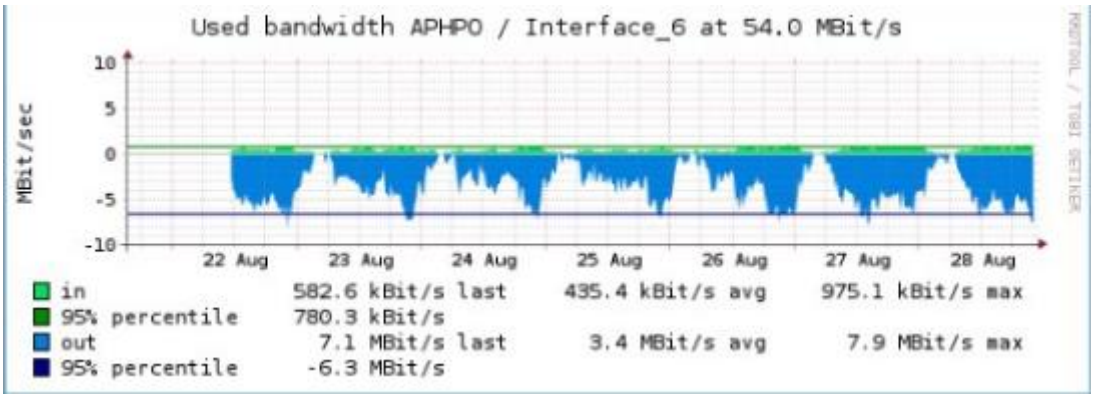


Figura 36 - OMD: Gráfico com o tráfego de uma interface *wireless*
Fonte: Autoria própria

Por todos esses resultados expostos, deu-se ao OMD a qualificação **boa**, pois atende todos os requisitos exigidos para tal. Por esta qualificação, o OMD ocupou a primeira colocação neste quesito.

5.3.2.2 Cacti

Assim como o OMD, os gráficos do Cacti também são gerados e armazenados pelo RRDTool, porém todos os gráficos pretendidos devem ser criados individualmente após a adição do dispositivo. Após adicionar um novo dispositivo ao Cacti, no canto superior direito aparecerá um link para a criação dos gráficos para aquele dispositivo, selecionando esta opção, o administrador encontrará a relação de quais são os serviços disponíveis para geração dos gráficos, basta selecionar os serviços e o tipo do gráfico a ser gerado e clicar em "*create*".

A partir deste momento, o Cacti já iniciará a geração dos gráficos, que podem ser visualizados no menu "*Graphs*". Para esta análise, foram monitorados a latência de todas as estações de rádio, o tráfego de todas as interfaces do *switch* e também 3 serviços da estação de gerência (consumo de memória, usuários conectados e quantidade de processos).

O Cacti ainda permite a personalização de todos os gráficos gerados, como cores, tipo do gráfico (área, linha, barras, etc), títulos e legendas. Essa personalização é feita através do menu "*Graph templates*", o usuário deve selecionar o *template* que deseja editar e terá à disposição uma página com todas as opções de personalização disponíveis. Para este trabalho, a única alteração realizada nos gráficos foi a tradução dos títulos e legendas e a adição dos objetos *ifAlias* e *ifName* como título dos gráficos de tráfego das interfaces do *switch*.

O Cacti permite a seleção dos gráficos por períodos pré-estabelecidos ou períodos personalizados, a exibição de miniaturas dos gráficos e também permite zoom em qualquer um dos gráficos gerados. As Figuras 37, 38, 39 e 40 exibem, respectivamente, o gráfico gerado com o tráfego da interface gigabit do *switch*, o gráfico de uma interface megabit do *switch*, a latência de uma das estações *wireless* e os gráficos gerados para a estação de gerência durante o período da análise (consumo de memória, quantidade de usuários conectados e quantidade de processos em execução).

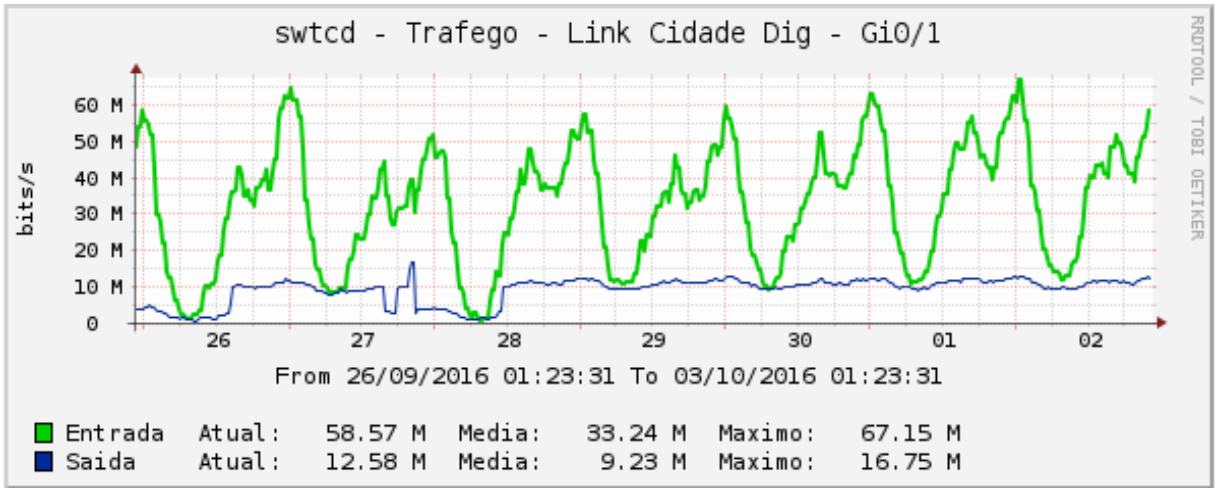


Figura 37 - Cacti: Gráfico da interface gigabit do *switch*
Fonte: Autoria própria

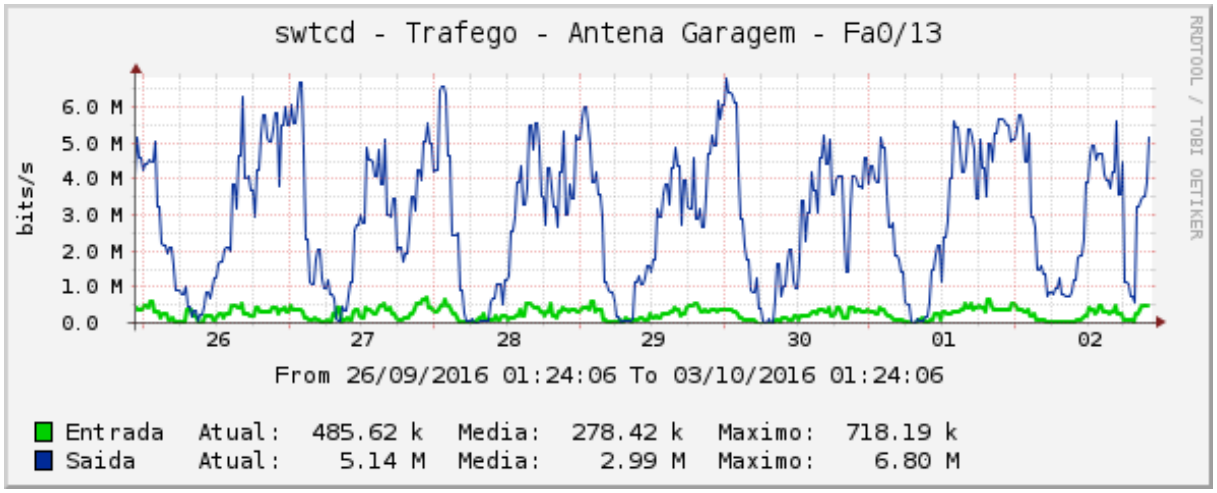


Figura 38 - Cacti: Gráfico de uma interface megabit do *switch*
Fonte: Autoria própria

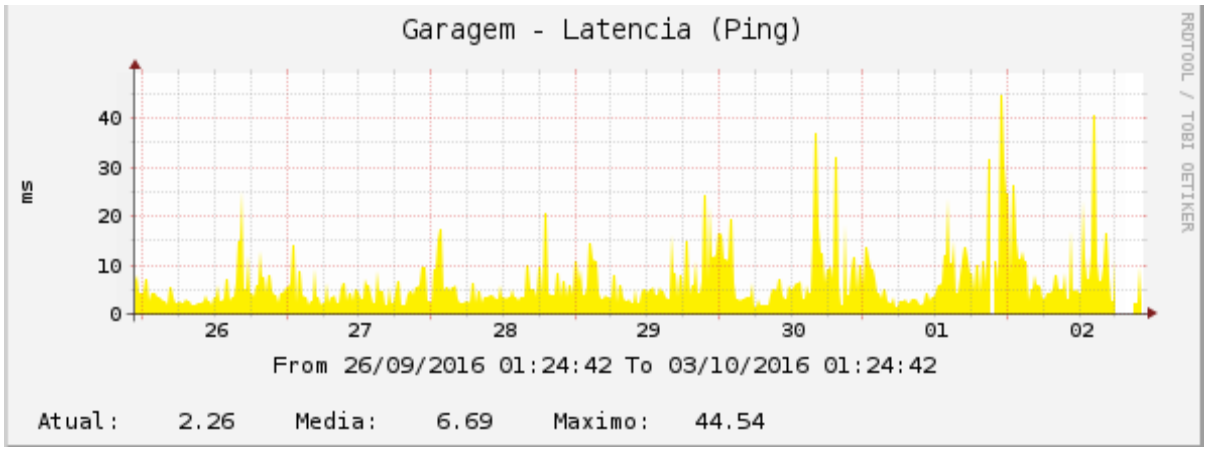


Figura 39 - Cacti: Gráfico com a latência de uma estação *wireless*
Fonte: Autoria própria

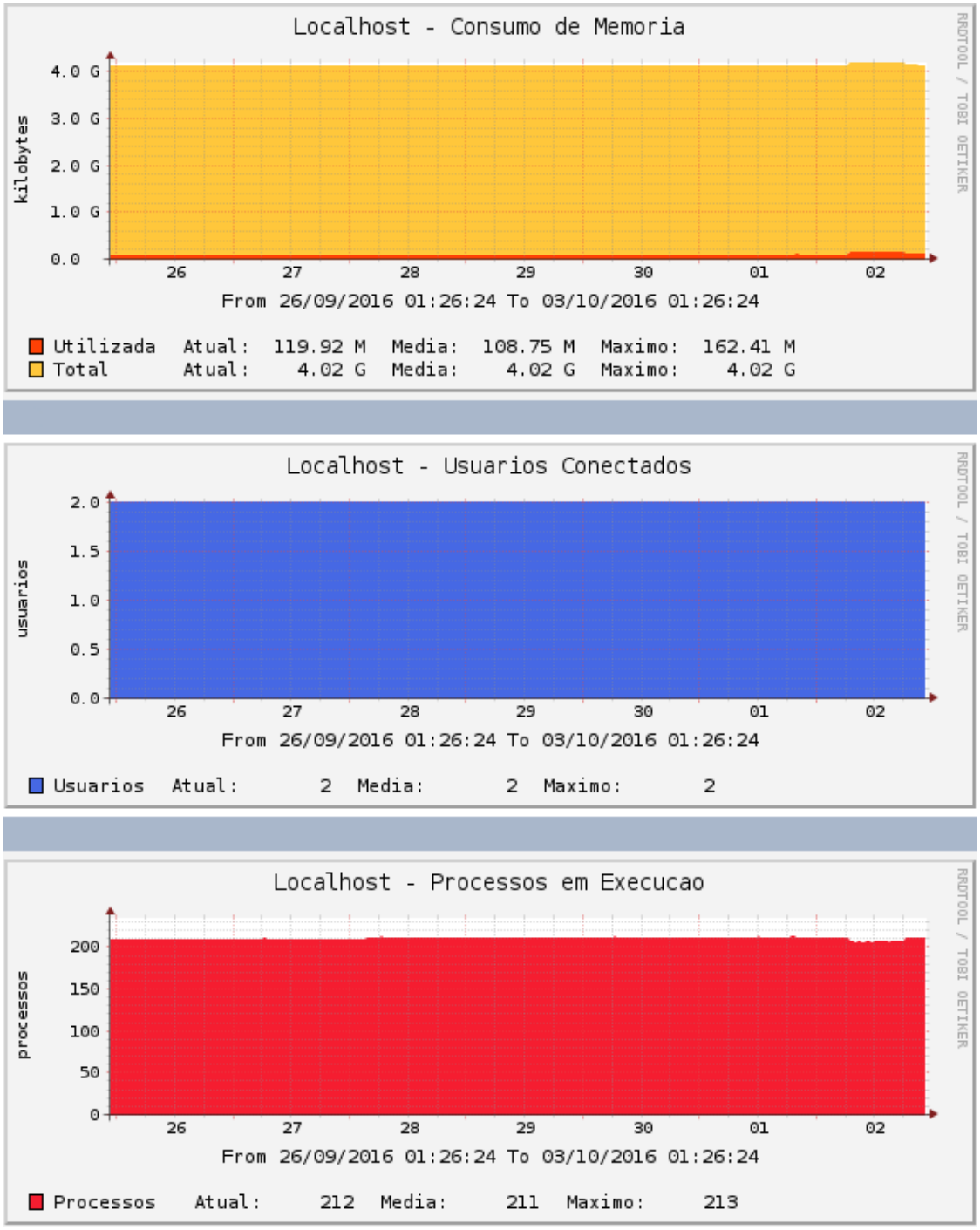


Figura 40 - Cacti: Gráficos gerados para a estação de gerência
Fonte: Autoria própria

Pelos fatos apresentados nesta análise, o Cacti recebeu como avaliação o atributo “**boa**” no que se refere a geração dos gráficos. Essa atribuição posicionou a ferramenta na primeira colocação.

5.3.2.3 Zabbix

Diferentemente das outras duas ferramentas analisadas neste trabalho, a geração de gráficos no Zabbix não é feita por ferramentas desenvolvidas por terceiros, sendo este recurso produzido pela própria ferramenta.

Ao finalizar a adição de um novo *host*, não necessariamente os gráficos estarão sendo gerados. Assim como as *triggers*, os gráficos do Zabbix também estão atrelados aos *templates*. Caso o *template* não possua gráficos, os mesmos podem ser criados pelo administrador e vinculados diretamente ao *host* ou ao *template*. Durante esta análise, o *template* escolhido para o *switch* e também para a estação de gerência já possuíam gráficos vinculados, sendo gerados 27 e 16 gráficos respectivamente a cada dispositivo. O *template* ICMP Ping utilizado para o monitoramento das estações de rádio não possuía gráficos, por este motivo, foi necessária a criação de um gráfico para ser vinculado ao *template*.

Para criar um novo gráfico atrelando-o a um *template*, o administrador deve acessar a opção "*Configuration" > "Templates"*, e selecionar a opção "*Graphs*" do *template*. A página seguinte apresenta a opção "*Create graph*"; clicando nesta opção o cadastro para novo gráfico exige algumas informações, sendo que as que foram modificadas para a criação do gráfico estão citadas abaixo:

- *Name*: nome dado ao gráfico a ser criado, aqui foi utilizada a nomenclatura "Ping";
- *Graph type*: tipo do gráfico, como, por exemplo, gráfico de pizza ou de barras. Neste caso foi utilizado o gráfico do tipo "normal";
- *Items*: nesta opção, deve ser selecionada qual *trigger* irá entregar os dados ao gráfico para que o mesmo os apresente. Para gerar o gráfico com o tempo de resposta, foi utilizada a *trigger* padrão do *template* chamada "ICMP *response time"*, que apresenta o tempo de resposta do dispositivo.

Ao término desta configuração, o gráfico criado foi atrelado ao *template* ICMP Ping.

Todos os gráficos gerados pelo Zabbix possuem opções de tempo pré-configurados para: 5, 15 ou 30 minutos; 1, 2, 3, 6 ou 12 horas; 1, 3 ou 7 dias. Além disso, é permitido zoom personalizado selecionando parte do gráfico e exibindo os valores mínimos, máximos e a média obtida durante o período de tempo escolhido.

As Figuras 41 e 42 apresentam respectivamente, o gráfico gerado com o tráfego da interface gigabit do *switch* e o gráfico de uma interface megabit, durante o período da análise.

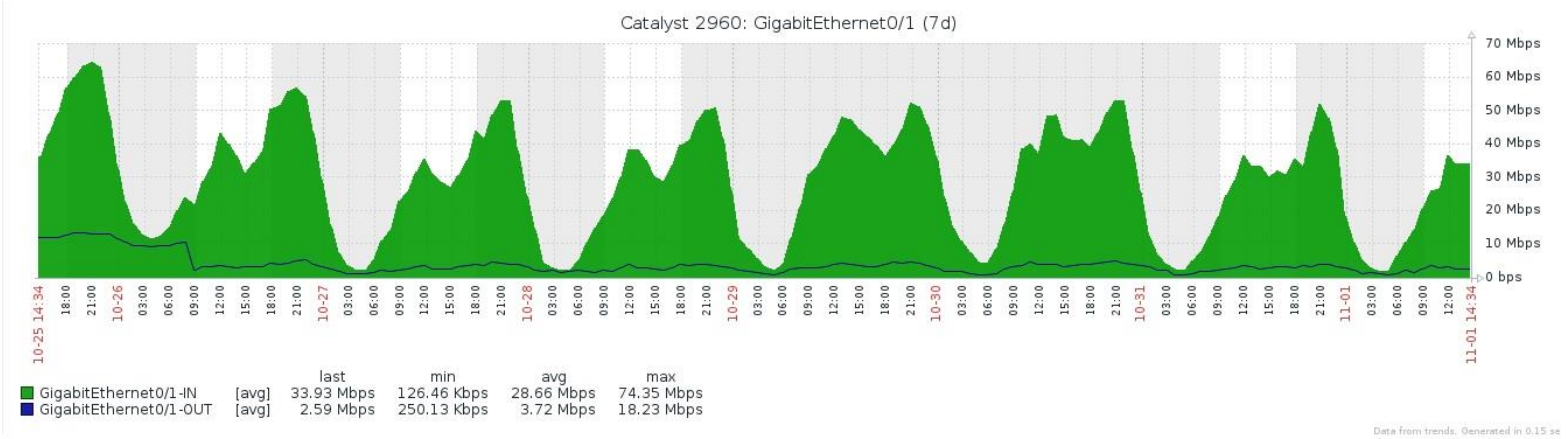


Figura 41 - Zabbix: Gráfico da interface gigabit do *switch*
Fonte: Autoria própria

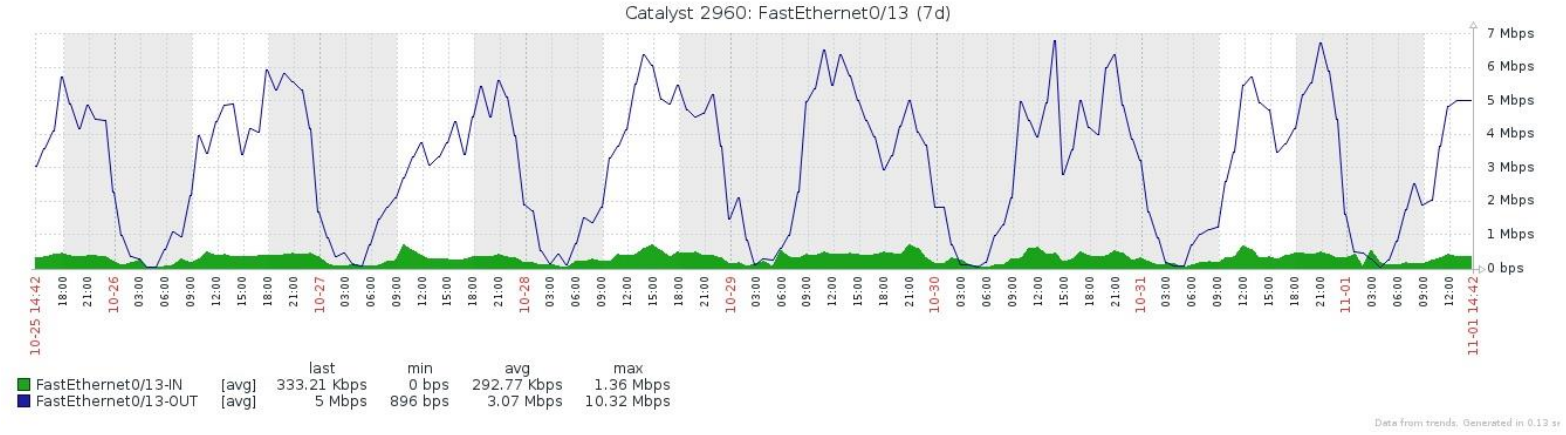


Figura 42 - Zabbix: Gráfico de uma interface megabit do switch
Fonte: Autoria própria

As Figuras 43 e 44 apresentam o gráfico criado anteriormente para exibir a latência das estações de rádio e alguns gráficos gerados com informações a respeito da estação de gerência.

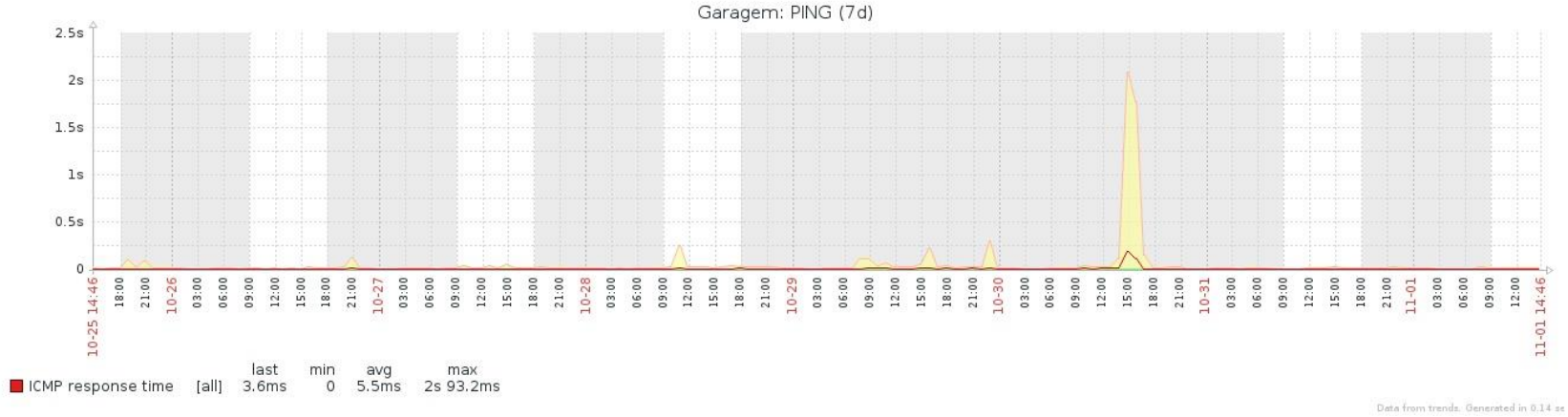


Figura 43 - Zabbix: Gráfico com a latência de uma estação de rádio
Fonte: Autoria própria

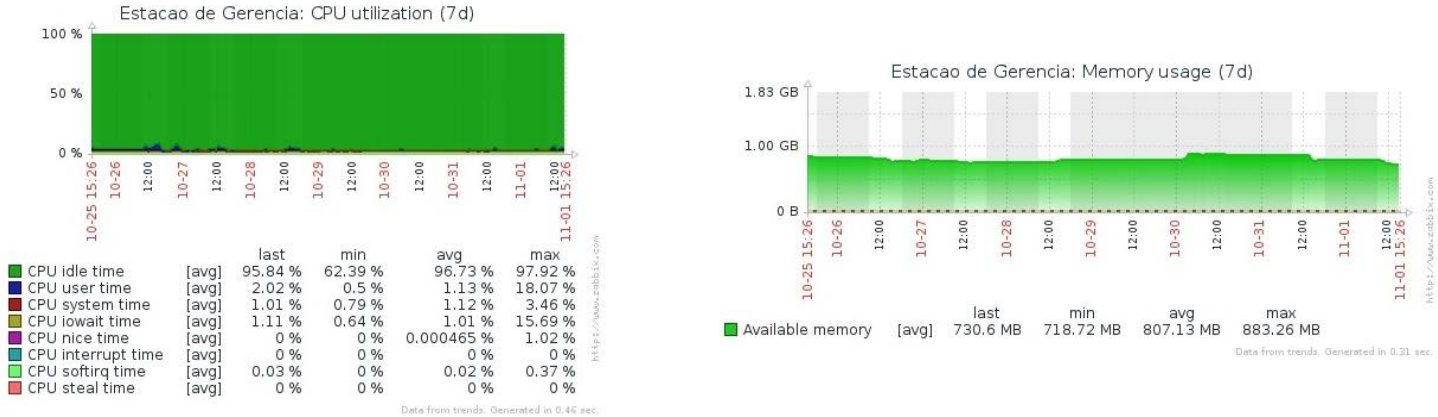


Figura 44 - Zabbix: Gráficos de utilização de CPU e memória na estação de gerência
Fonte: Autoria própria

Levando em conta todos os fatos aqui postos, o Zabbix também recebeu avaliação **boa** para a geração de gráficos, o que manteve todas as ferramentas com avaliação máxima neste quesito.

O Quadro 12 exibe o resultado das avaliações referente a geração de gráficos.

Ferramenta	**Geração de gráficos**	**Posição da Ferramenta**
OMD	Boa	1ª colocação
Cacti	Boa	1ª colocação
Zabbix	Boa	1ª colocação

Quadro 12 - Resultado: Geração de gráficos
Fonte: Autoria própria

5.3.3 Notificações

É fundamental que uma ferramenta de monitoramento possua recursos para notificar o administrador acerca de determinados eventos que possam ocorrer na rede. Nesta Seção são apresentadas as avaliações e resultados sob os critérios estabelecidos na Seção 4.2.2.3.

5.3.3.1 OMD

Assim como outras configurações realizadas através do Check_MK, as relativas às notificações também são acessadas através do WATO. Acessando o menu “*Notifications*”, o administrador encontra uma interface com as regras de notificações; por padrão, já há uma regra configurada para a pasta *main*, porém ela não gera notificações.

Para o desenvolvimento deste trabalho foi utilizada apenas uma regra, uma vez que não há maiores necessidades dentro do cenário estudado. Ao clicar para editar a regra, o administrador é apresentado a uma interface que possui diversas opções; destacam-se as relativas ao método de notificação e as condições. Para esta análise foi utilizado o método de HTML E-mail, porém, a ferramenta oferece outras opções, como, por exemplo, envio de SMS (*Short Message Service*). Utilizando HTML E-mail, o método de notificação solicita informações como: assunto

do e-mail e quais informações aparecerão no corpo do e-mail, além de outras personalizações que podem ser incluídas pelo administrador.

As configurações realizadas para esta análise foram para que os assuntos dos e-mails apresentassem o *hostname*, o nome do serviço (caso a notificação seja de um serviço) e o evento ocorrido, e que o corpo dos e-mails apresentasse o endereço IP do *host*, o horário do evento, os dados obtidos do *host* e o gráfico gerado. A Figura 45 apresenta essas regras.

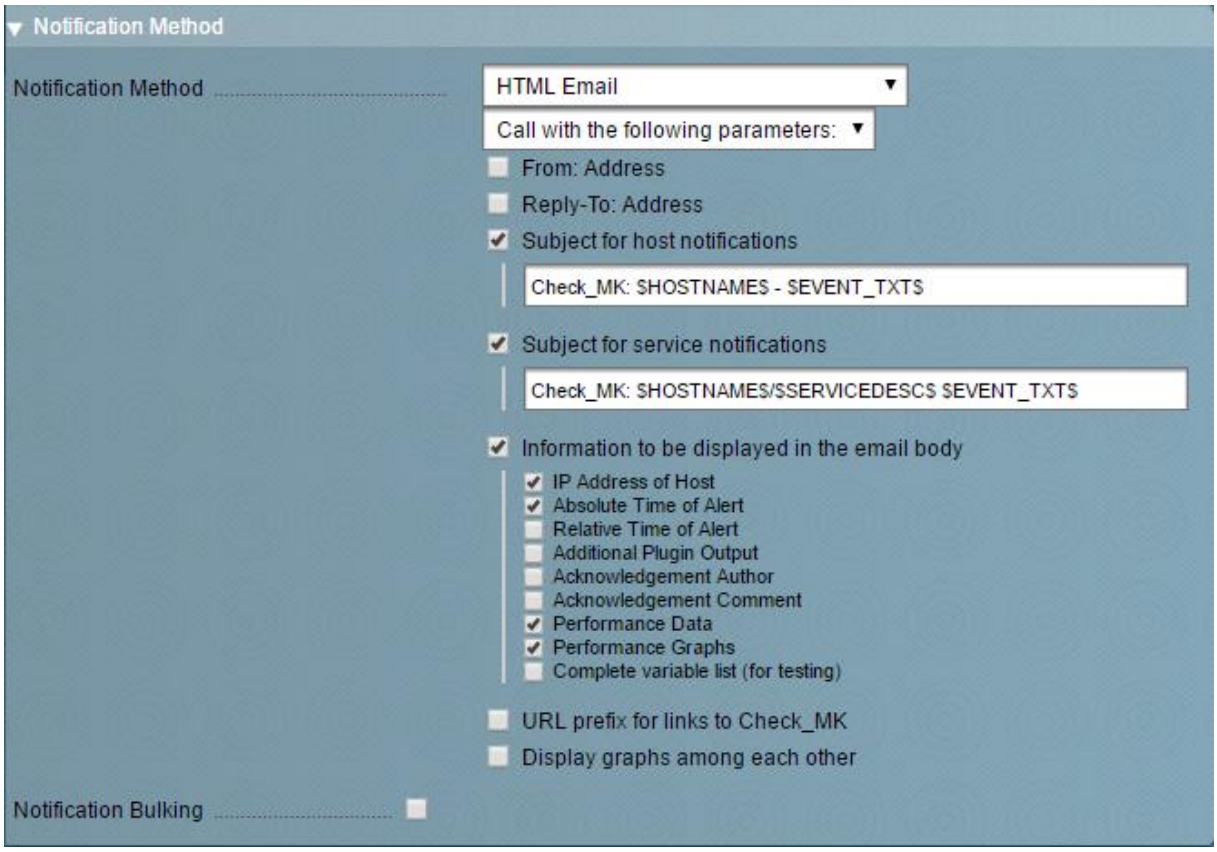


Figura 45 - OMD: Regras de notificações
Fonte: Autoria própria

Abaixo das configurações que dizem respeito ao método de notificação há a possibilidade de explicitar determinados usuários para determinados *hosts*, ou então explicitar quais e-mails receberão as notificações. Por haver apenas um usuário cadastrado, as notificações de todos os *hosts* foram enviadas a todos os usuários.

Outra configuração que deve ser feita para habilitar corretamente as notificações é a definição de quais serão os eventos que irão gerar notificações aos usuários.

A Figura 46 apresenta como ficaram as configurações dos eventos.

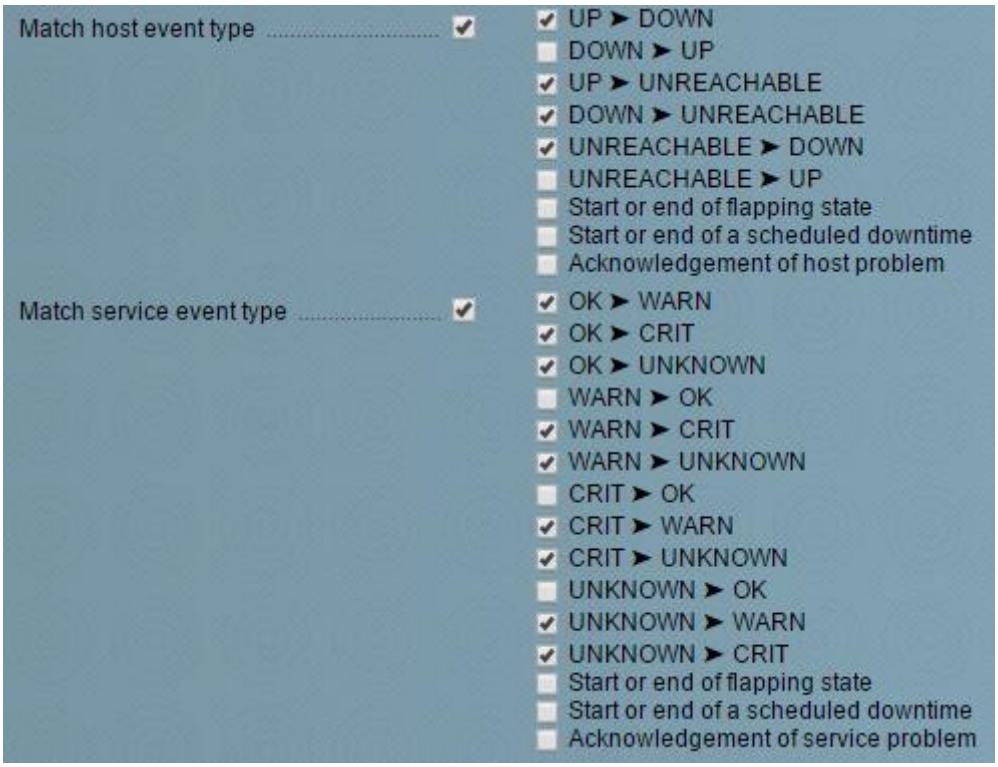


Figura 46 - OMD: Regras para os eventos de notificações
Fonte: Autoria própria

A última configuração que deve ser realizada para habilitar as notificações é a atribuição dos usuários a um grupo de contato. Isso também deve ser feito através do WATO, mas, por padrão, os usuários criados já pertencem ao grupo de contato "*Everybody*", portanto, neste caso não foi necessária nenhuma alteração. Com essas configurações realizadas, o OMD, em conjunto com o Check_MK, já está hábil a enviar notificações. O exemplo de uma notificação que ocorreu em uma estação de rádio enquanto estava sendo realizada a análise pode ser vista na Figura 47.

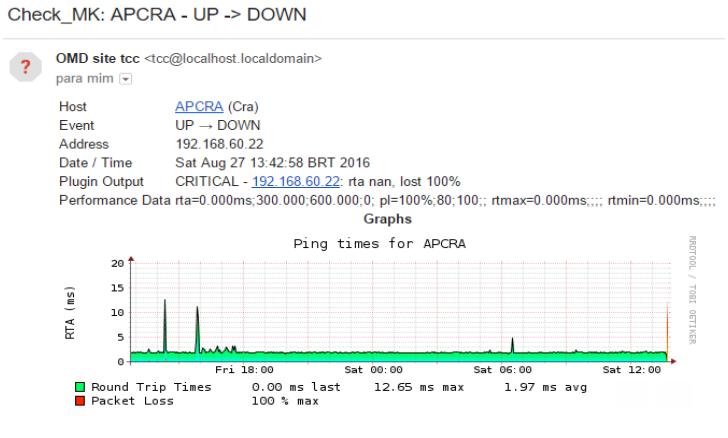


Figura 47 - OMD: Notificação via *e-mail* do evento um *host*
Fonte: Autoria própria

Pode-se perceber que todas as informações selecionadas para serem apresentadas no e-mail foram devidamente apresentadas. Assim, percebe-se que o *host* APCRA teve um evento de "*up*" para "*down*", no dia 27 de agosto às 13:42:58, ficando em estado crítico, pois teve uma perda de 100% de seus pacotes.

Por todos estes fatos apresentados, deu-se ao OMD a avaliação **sim**, já que a ferramenta possui nativamente o recurso de notificações, sendo que o mesmo ainda se mostra eficiente, totalmente personalizável e de fácil configuração. Tal motivo coloca a ferramenta na primeira colocação.

5.3.3.2 Cacti

Assim como os mapas e diagramas de rede, o Cacti **não** possui o recurso de notificações de forma nativa; este recurso só é possível através do uso de *plug-ins* desenvolvidos pela comunidade. Como citado anteriormente, recursos externos à ferramenta não foram levados em consideração por esta análise. Por este motivo, o Cacti ficou na segunda colocação neste quesito.

5.3.3.3 Zabbix

A configuração do envio de notificações sobre eventos da rede através de *e-mail* utilizando o Zabbix é feita através de três configurações distintas, tendo como pré-requisito a utilização de um servidor de e-mail na estação de gerência, neste caso foi instalada e configurada a ferramenta PostFix.

A primeira etapa de configuração no Zabbix é a criação de um "*Media type*", que representa a forma que o Zabbix utilizará para enviar as notificações. Neste caso foi criado um "*media type*" do tipo e-mail, no entanto, vale destacar que o Zabbix aceita outras opções, como, por exemplo, a utilização de *scripts* externos e uso de SMS. O formulário para configuração de um *media type* é encontrado no menu "*Administration*" > "*Media types*". Nesta etapa devem ser inseridas as configurações do servidor de e-mail previamente configurado na estação de gerência.

A Figura 48 apresenta a página de configuração de um *media type.* Por questões de segurança o endereço IP da estação de gerência e e-mail utilizado para envio das mensagens foram ocultados.

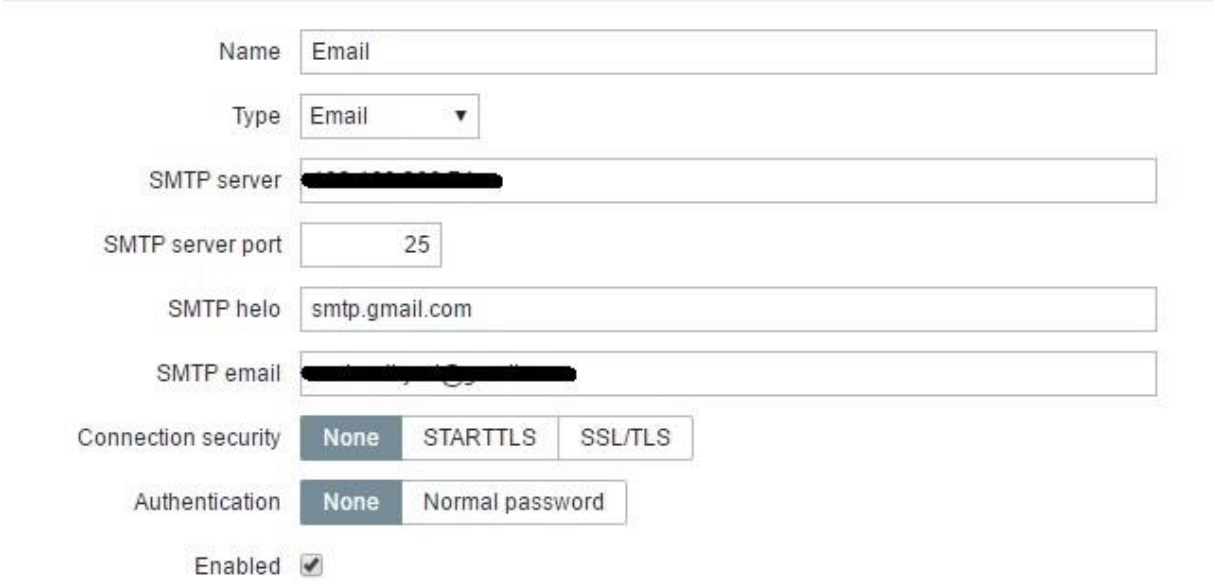


Figura 48 - Zabbix: Novo media type para o envio de notificações via e-mail
Fonte: Autoria própria

As informações modificadas para a criação do *media type* foram as seguintes:

- *Name*: nome dado ao *media type* que está sendo configurado;
- *Type*: tipo do *media type*, como já citado, neste caso foi utilizado o tipo “e-mail”;
- *SMTP (Simple Mail Transfer Protocol) Server*: endereço IP do servidor SMTP, neste caso, o próprio *localhost*;
- *SMTP Server Port*: porta utilizada pelo servidor SMTP, por padrão a porta 25;
- *SMPT* helo: helo utilizado pelo SMTP. Como o e-mail utilizado para envio das mensagens através do PostFix foi um endereço do Gmail, utilizou-se o helo do Gmail (smtp.gmail.com);
- *SMTP email:* e-mail configurado no PostFix para envio das mensagens.

Além dessas configurações, o Zabbix permite que sejam informados (quando necessário) métodos de autenticação e/ou uso de padrões de segurança como o SSL (*Secure Socket Layer*), não utilizados neste cenário.

Após a configuração do *media type*, o administrador deve configurar a conta do usuário para a qual serão enviadas as notificações. Para isso é necessário acessar o menu "*Administration"* > *"Users"*, e deve ser selecionado o usuário que irá receber a configuração, neste caso, o usuário "Admin". Na aba "*Media"* deve ser adicionada uma nova *media* do tipo e-mail, preenchendo o endereço de e-mail que deverá receber as notificações.

A última configuração exigida para que de fato o Zabbix envie as notificações via e-mail é a criação de uma *action* que irá disparar o e-mail quando houver o acontecimento de determinado evento. A *action* criada é semelhante à criada para a adição dos *hosts* após a autodescoberta, porém, alteram-se as condições e operações. Para que haja o envio de e-mails, as condições devem ser *triggers* vinculadas aos *hosts* e a operação deve ser o envio de uma mensagem ao usuário que possui a conta de e-mail configurada; neste caso foram configuradas as *triggers* que representam que um *host* está indisponível via ICMP Ping.

O resultado de um e-mail enviado pelo Zabbix pode ser visualizado na Figura 49.

Figura 49 - Zabbix: Notificação via e-mail do evento um host
Fonte: Autoria própria

A notificação enviada pelo Zabbix é mais simples e possui menos informações do que a proveniente do OMD, por exemplo, todavia, auxilia o administrador para saber em tempo real sobre determinados eventos na rede. Além disso, o Zabbix atende aos requisitos exigidos por esta avaliação, por isso, atribuiu-se **sim** à esta métrica. Por este resultado, o Zabbix ocupou a primeira colocação neste quesito, juntamente com o OMD.

O Quadro 13 exibe o resultado das avaliações referente a capacidade nativa da ferramenta em gerar notificações sobre eventos da rede ao administrador.

Ferramenta	Notificações	Posição da Ferramenta
OMD	Sim	1ª colocação
Zabbix	Sim	1ª colocação
Cacti	Não	2ª colocação

Quadro 13 - Resultado: Notificações
Fonte: Autoria própria

5.4 RESULTADO FINAL

O Quadro 14 apresenta um resumo com todas as métricas analisadas, os resultados obtidos e a colocação da ferramenta diante da respectiva métrica.

Critério avaliativo	**Métrica**	**OMD**	**Cacti**	**Zabbix**	**1ª colocação**	**2ª colocação**	**3ª colocação**
Instalação das ferramentas	Grau de dificuldade durante a instalação	Grau II	Grau III	Grau I	Zabbix	OMD	Cacti
	Necessidade de pacotes adicionais	61 pacotes	48 pacotes	39 pacotes	Zabbix	Cacti	OMD
	Tempo de instalação	28 minutos	24 minutos	18 minutos	Zabbix	Cacti	OMD
Configuração dos *hosts* e serviços	Autodescoberta de *hosts*	Não	Não	Sim	Zabbix	Cacti/OMD	
	Criação de regras de monitoramento	Sim	Não	Sim	OMD/Zabbix	Cacti	
	Agente próprio	Sim	Não	Sim	OMD/Zabbix	Cacti	
	Grau de dificuldade na configuração	Grau I	Grau I	Grau I	OMD/Cacti/Zabbix		
Geração de informações	Mapas e diagramas de rede	Boa	Ruim	Regular	OMD	Zabbix	Cacti
	Geração de gráficos	Boa	Boa	Boa	OMD/Cacti/Zabbix		
	Notificações	Sim	Não	Sim	OMD/Zabbix	Cacti	

Quadro 14 - Resultado final
Fonte: Autoria própria

Analisando o Quadro 14, percebe-se que o Zabbix esteve na primeira colocação em nove das dez métricas analisadas. Em segundo lugar apresenta-se o OMD, com um total de seis primeiras colocações. O Cacti ficou com a terceira e última colocação, tendo conseguido apenas duas primeiras colocações dentre dez possíveis. Através deste resultado, conclui-se que, para o cenário em questão, a ferramenta mais adequada dentre as três analisadas foi o Zabbix, visto que esta alcançou um melhor índice geral ao final da análise.

Cabe ressaltar que este resultado se apresentou no cenário determinado e que as ferramentas podem comportar-se de maneiras diferentes em outros cenários. Também é importante frisar que as métricas subjetivas utilizadas dependiam da interpretação do autor deste trabalho, outros autores podem ter outras intepretações e consequentemente outros resultados.

6 CONCLUSÃO

A importância das redes de computadores na vida das pessoas está implícita diariamente, mesmo para aqueles cuja vida não está diretamente ligada às tecnologias de comunicação. Os seres humanos estão vivendo cada vez mais conectados, seja pelas mídias sociais, mensageiros instantâneos, trocas de *e-mails*, videoconferências ou qualquer outro meio utilizado para comunicação sob protocolos de Internet. Todo esse “mundo conectado” exige que as empresas fornecedoras de conteúdo garantam a qualidade máxima de seus produtos e serviços, porém, não basta um serviço de qualidade se o acesso ao mesmo for ineficaz e, nesta questão, entram os provedores de Internet, com a responsabilidade de, como o próprio nome já diz, prover acesso à Internet aos seus clientes.

Para prover um serviço com máxima qualidade e desempenho possível, é fundamental que os provedores saibam o que está acontecendo em suas redes e este é, justamente o objeto de estudo deste trabalho: as ferramentas de monitoramento e gerenciamento de redes de computadores. O presente trabalho teve o objetivo de encontrar a ferramenta mais adequada para monitoramento e gerenciamento de redes de computadores dentro do cenário exposto, o que foi alcançado através de análises minuciosas utilizando critérios e métricas pré-determinadas.

Ao final das análises realizadas neste estudo, o Zabbix foi determinado como a solução mais adequada ao cenário apresentado, neste caso, a rede do programa Cidade Digital do Município de São Lourenço do Oeste. Cabe ressaltar que as três ferramentas analisadas possuem o mesmo objetivo, todavia cada uma com as suas características. Destaca-se também que este estudo não teve a pretensão de definir qual era a melhor ferramenta, mas sim, qual se comportou de forma mais adequada durante o período de análise no referido cenário.

O estudo ainda auxiliou o provedor de Internet, no sentido de estudo de caso sobre a rede, já que pode-se tirar como conclusão, diante dos dados provenientes da análise, que a rede em questão não possui grandes gargalos, principalmente no que compete ao *throughput* do *uplink* e o *throughput* das estações de rádio. Também foi possível identificar, através do monitoramento das interfaces do *switch*, que algumas estações de rádio estão subutilizadas, tendo pouquíssimo tráfego mesmo em períodos onde há maior número de usuários conectados.

Este estudo também contribuiu para o desenvolvimento pessoal do autor, visto que ampliou seus conhecimentos através do trabalho multidisciplinar implementado, possibilitando a aplicação de diversos conteúdos apreendidos ao longo do curso. O presente estudo ainda poderá auxiliar administradores de redes que necessitem de um estudo detalhado sobre as ferramentas analisadas, além de poder auxiliá-los nas tarefas de implantação e configuração das mesmas.

6.1 TRABALHOS FUTUROS

Destacam-se, a seguir, os trabalhos futuros a serem executados pelo autor, aproveitando os estudos obtidos neste trabalho.

- Análise de outros fatores das ferramentas, como, por exemplo, o consumo do recurso de hardware (CPU, memória, disco rígido e rede);
- Levar em consideração os *plug-ins* existentes nas comunidades das ferramentas, principalmente para o Cacti e o Zabbix, já que estes possibilitam maior aproveitamento dos recursos das ferramentas;
- Implantação da ferramenta escolhida no ambiente do provedor de forma definitiva, já que as três ferramentas foram removidas do cenário após o estudo;
- Inclusão dos outros dispositivos que fazem parte da rede abordada e que não foram monitorados durante este estudo (*WebFilter,* MikroTik RouterBoard e o servidor *Radius*).

REFERÊNCIAS

BALBINOT, Luís F.; ANDRADE, Maiko. **Uma ferramenta flexível para a medição de tráfego baseada no RRDtool**. Julho de 2000. Disponível em: <https://memoria.rnp.br/newsgen/0007/art7.html>. Acesso em: 15 set. 2016.

BARRIVIERA, Rodolfo Msc. **Gerência de Redes de Computadores:** Protocolo CMIP, 2010, Instituto Federal do Paraná – Campus Londrina. Disponível em: <http://www.rodolfobarriviera.com.br/arquivos/disciplinas/40_arquivo.pdf>. Acesso em: 15 mai. 2016.

BERNAL, Huber F. **Simple Network Management Protocol (SNMP)**. Janeiro de 2014. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialsnmp/pagina_1.asp> Acesso em: 18 mai. 2016.

CACTI. **Cacti – The Complete RRDTool-Base Graphing Solution**. Disponível em: <http://www.cacti.net/what_is_cacti.php>. Acesso em: 22 mai. 2016.

CASE, J et al. **A Simple Network Management Protocol (SNMP)**, RFC1157, Maio 1990. Disponível em: <https://www.ietf.org/rfc/rfc1157.txt>. Acesso em: 09 maio 2016.

CISCO. **VPN:** Conexão segura a escritórios, usuários e parceiros. Disponível em: < http://www.cisco.com/web/BR/solucoes/pt_br/vpn/index.html>. Acesso em: 21 mai. 2016.

DOUGLAS, Mauro R.; SCHMIDT, Kevin J. **Essential SNMP**, 2nd ed, Sebastopol, O'Reilly Media, 2005.

ELER, Esdras de O. **Modelo TMN: Aplicação ao Gerenciamento de Redes de Telecomunicações**, Out. 2015. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialmodelotmn>. Acesso em: 29 abr. 2016.

FOROUZAN, Behrouz A. **Comunicação de Dados e Redes de Computadores**, 3.ed, Tradução de Glayson Eduardo de Figueiredo, Porto Alegre, Bookman, 2006.

_____. **Data Communications and Networking**, 4th ed, New York, McGraw Hill, 2007.

_____; MOSHARRAF, Firouz. **Redes de Computadores:** Uma Abordagem Top-Down. 1. ed. Tradução de Marcos A. Simplicio Jr e Charles Christian Miers, Porto Alegre, AMGH Editora Ltda, 2013.

HERTZOG, Raphaël; MAS, Roland. **O Manual do Administrador Debian:** Debian Jessie, da Descoberta à Maestria. 1 ed, 2015.
ISO/IEC/IEEE 24765. **Systems and software engineering – Vocabulary**. 1st ed., Dezembro de 2010.

LEHMANN, Erny O. **Especificação e Verificação do Protocolo CMIP para Gerenciamento de Redes**. Tese (Mestrado em Teleinformática), Universidade Federal do Rio de Janeiro. Disponível em <http://www.gta.ufrj.br/grad/cmip.html>. Acesso em: 15 mai. 2016.

LOBÃO, Henrique F. **Comparativo entre as versões do SNMP**. Setembro de 2011. Disponível em <http://www.fassi.eti.br/artigos/gerencia-de-redes/comparativo-entre-as-versoes-do-snmp>. Acesso em: 23 mai. 2016.

LOPES, Raquel V.; SAUVÉ, Jacques P.; NICOLLETTI, Pedro S. **Melhores Práticas Para Gerência de Redes de Computadores**. 1. ed. Editora Campus, 2002.

MOURA, Alex. **Gerenciamento de Redes com Software Livre**, Out. 2005. Disponível em: <https://memoria.rnp.br/_arquivo/sci/2005/moura-alex_gerenciamento-redes.pdf>. Acesso em: 02 abr. 2016.

OMD. **OMD – The Open Monitoring Distribution**. Disponível em: <http://omdistro.org>. Acesso em: 22 mai. 2016

POLETO, Olavo F. **Gerenciamento e Monitoramento de Redes II: Análise de Desempenho**, Jan. 2012. Disponível em: <http://www.teleco.com.br/tutoriais/tutorialgmredes2/pagina_2.asp>. Acesso em: 14 mai. 2016.

SATO, Danilo T. **Uso eficaz de métricas em métodos ágeis de desenvolvimento de software**. 2007. 139 f. Dissertação (Mestrado em Ciência) – Instituto de Matemática e Estatística. Universidade De São Paulo, São Paulo, 2007.

SPECIALSKI, Elizabeth. S. **Gerência de Redes de Computadores e de Telecomunicações**, Universidade Federal de Santa Catarina, Florianópolis, 1999.

STALLINGS, William. **SNMP, SNMPv2, SNMPv3 and RMON 1 and 2**, 3.ed, [S.I.], Addisson-Wesley, 1999.

VLADISHEV, Alexei. **Zabbix – Monitoring Solution for Everyone**. Paris Zabbix User Group Meetup 2016, 2016, Paris. Disponível em: <http://www.slideshare.net/Zabbix/alexei-vladishev-zabbix-monitoring-solution-for-everyone>. Acesso em: 15 out. 2016.

ZABBIX. **Zabbix – The Enterprise-class Monitoring Solution for Everyone**. Disponível em: <https://www.zabbix.com/documentation/3.0>. Acesso em: 14 out. 2016.

4Linux. **O que é Zabbix**. Disponível em: <https://www.4linux.com.br/o-que-e-zabbix>. Acesso em: 14 out. 2016.

APÊNDICES

APÊNDICE A – yum install --nogpgcheck omd-1.30-el7-35.x86_64rpm

APÊNDICE B – omd create tcc; su -tcc; omd start

APÊNDICE C – /usr/sbin/setsebool httpd_can_network_connect 1

APÊNDICE D – cp /usr/lib64/python2.7/hashlib.py /omd/versions/1.30/lib/python/hashlib.py

APÊNDICE E – yum install check-mk-agent-1.2.6p12-1.noarch.rpm

APÊNDICE F – yum install httpd httpd-devel mariadb-server php-mysql php-pear php-common php-gd php-devel php php-mbstring php-cli php-snmp net-snmp-utils net-snmp-libs rrdtool

APÊNDICE G – systemctl start httpd.service mariadb.service snmpd.service; systemctl enable httpd.service mariadb.service snmpd.service

APÊNDICE H – mysqladmin -u root password asd123; mysql -u root -p; create database cacti; GRANT ALL ON cacti.* TO cacti@localhost IDENTIFIED BY ‘asd123’; FLUSH privileges; quit;

APÊNDICE I – yum install cacti

APÊNDICE J – mysql -u cacti -p cacti < /usr/share/doc/cacti-0.8.8h/cacti.sql

APÊNDICE K – firewall-cmd –permanent –zone=public –add-service=http

APÊNDICE L – yum install httpd mariadb mariadb-server php php-pear php-gd php-mysql

APÊNDICE M – rpm -ivh http://repo.zabbix.com/zabbix/3.0/rhel/7/x86_64/zabbix-release-3.0-5.el7.noarch.rpm

APÊNDICE N – yum install zabbix-server-mysql zabbix-web-mysql zabbix-agent

APÊNDICE O – mysql -uroot -p asd123; create database zabbix character set utf8 collate utf8_bin; grant all privileges on zabbix.* to zabbix@localhost identified by 'asd123'; quit; cd database/mysql; mysql -uzabbix -p asd123 zabbix < schema.sql; mysql -uzabbix -p asd123 zabbix < images.sql; mysql -uzabbix -p asd123 zabbix < data.sql; exit; zcat /usr/share/doc/zabbix-server-mysql-3.0.5/create.sql.gz | mysql -uroot -p zabbix