UNIVERSIDADE DO EXTREMO SUL CATARINENSE - UNESC

CURSO DE CIÊNCIA DA COMPUTAÇÃO

SIVALDO MARTINS

SIMULAÇÃO DE UTILIZAÇÃO DA FERRAMENTA DE GERÊNCIA DE REDES ZABBIX PARA MONITORAMENTO DE ATIVOS E ANÁLISE DE TENDÊNCIAS NO AUXÍLIO À ADMINISTRAÇÃO DE AMBIENTES DISTRIBUÍDOS

CRICIÚMA

2015

SIVALDO MARTINS

SIMULAÇÃO DE UTILIZAÇÃO DA FERRAMENTA DE GERÊNCIA DE REDES ZABBIX PARA MONITORAMENTO DE ATIVOS E ANÁLISE DE TENDÊNCIAS NO AUXÍLIO À ADMINISTRAÇÃO DE AMBIENTES DISTRIBUÍDOS

Trabalho de Conclusão de Curso, apresentado para obtenção do grau de Bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC.

Orientador: Prof. MSc. Paulo João Martins

CRICIÚMA

2015

SIVALDO MARTINS

SIMULAÇÃO DE UTILIZAÇÃO DA FERRAMENTA DE GERÊNCIA DE REDES ZABBIX PARA MONITORAMENTO DE ATIVOS E ANÁLISE DE TENDÊNCIAS NO AUXÍLIO À ADMINISTRAÇÃO DE AMBIENTES DISTRIBUÍDOS

Trabalho de Conclusão de Curso, aprovado pela Banca Examinadora para obtenção do grau de Bacharel no curso de Ciência da Computação da Universidade do Extremo Sul Catarinense, UNESC, com Linha de Pesquisa em Redes de Computadores.

Criciúma 26 de novembro de 2015

BANCA EXAMINADORA

Prof. Paulo João Martins – Mestre – (UNESC) Orientador

Prof. Rogério Antônio Casagrande – Mestre – (UNESC)

Prof. Valter Blauth Junior – Especialista – (UNESC)

AGRADECIMENTOS

Agradeço aos meus pais e minha irmã por acreditarem em mim e me incentivarem para a obtenção dessa conquista. Agradeço também a meu orientador Paulo João Martins, pela paciência e incentivo ao longo do projeto e também aos demais professores do curso de Ciência da Computação da Unesc.

“A persistência é o caminho do êxito.”

Charles Chaplin

RESUMO

A popularização das redes de computadores passou a influenciar a vida pessoal e profissional, houve diminuição dos custos de comunicação e aumento na produtividade das empresas. Como consequência ocorreu um crescimento na complexidade para administrá-las e manter todos os serviços ativos. O gerenciamento dessa estrutura ficou inviável de ser feito manualmente, e o emprego de softwares como apoio na tarefa passou a ser necessário. Existem muitos capazes de auxiliar um administrador na tarefa de gerenciar uma rede, sendo necessário pesquisá-los para saber qual melhor se ajusta a tarefa. Este projeto estudou e aplicou o uso do Zabbix como ferramenta de apoio ao administrador no gerenciamento de redes distribuídas, por meio da detecção de falhas e comportamentos não previstos capazes de gerar problemas. O estudo foi realizado em um ambiente controlado idealizado para testes com simulação de erros e eventos de rede, podendo-se então verificar a eficácia da ferramenta no apoio às decisões tomadas pelo administrador na aplicação dos conceitos de gerência. Durante o estudo procurou-se entender o ambiente e a ferramenta com o intuito de se obter um conhecimento mais amplo na implementação. Os testes demonstraram que a aplicação possui muitas funções capazes de auxiliar o administrador na função de gerenciar redes distribuídas.

Palavras-chave: Redes de Computadores. Gerenciamento. Zabbix.

ABSTRACT

The popularization of the computer networks began to influence the personal and professional life due to a communication costs reduction and the increased business productivity. As result occurred an increment in the complexity to manage them and maintain all the services actives. It was impossible to be done manually the management of this structure, and the use of the softwares to support the task has become necessary. There are many softwares to assist an administrator in the task of managing a network, that should be searched for finding out what best fits the task. This project studied and applied the use of the Zabbix as administrator support tool in the management of distributed networks, by fault detection and behaviors that generate unanticipated problems. The study was fulfilled in a controlled environment designed to tests with simulation of network errors and events. The effectiveness of the tool in supporting decisions made by the administrator in the management application concepts could be verified. During the study, we searched to understand the environment and the tool to obtain a wider knowledge during the implementation. The tests have shown that the application has many functions which can assist the administrator in charge of managing distributed networks.

Keywords: Computer networks. Management. Zabbix.

LISTA DE ILUSTRAÇÕES

LISTA DE ABREVIATURAS E SIGLAS

CPU	Central Process Unit
FTP	File Transfer Protocol
GPLv2	General Public Licence versão 2
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
ICMP	Internet Control Message Protocol
IMAP	Internet Message Access Protocol
IP	Internet Protocol
IPMI	Intelligent Platform Management Interface
ISO	International Standards Organization
JMX	Java Management Extensions
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
LTS	Long Term Support
MIB	Management Information Base
NETSTAT	Network Statistic
ODBC	Open Database Connectivity
PING	Packet Internet Groper
POP	Post Office Protocol
RAM	Random Access Memory
SSH	Secure Shell
SMS	Short Message Service
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SSH	Secure Shell
TCP	Transmission Control Protocol
TI	Tecnologia da Informação
UDP	User Datagram Protocol
VPN	Virtual Private Network
WAN	Wide Area Network

SUMÁRIO

1 INTRODUÇÃO

O uso da tecnologia nos ambientes corporativos no princípio da informatização das empresas, era baseado no uso de computadores de grande porte, a arquitetura da informação era centralizada e o gerenciamento simplificado.

A evolução dos periféricos e seu compartilhamento abre caminho para o surgimento das redes. Como consequência há a redução de custos e aumento de produtividade. As redes também evoluíram rapidamente em complexidade e robustez.

Essa revolução alterou nosso estilo de vida. Hoje as decisões tomadas, no âmbito pessoal e profissional, são influenciadas pelas informações disponíveis (FOROUZAN, 2006).

Para se obter essas informações a qualquer momento, é preciso de uma infraestrutura de tecnologia disponível o máximo de tempo possível. Esse alto índice de disponibilidade é obtido com controle dos elementos da rede, evitando que sejam subutilizados, ou usados para fins indevidos.

Manter os recursos funcionando tornou-se complexo, demandando cada vez mais tempo para uma análise mais detalhada sobre as ações que devem ser executadas na solução dos problemas enfrentados.

O uso de ferramentas, para a divisão da complexidade e auxilio na função de gerenciar uma rede local ou distribuída, passa a ser relevante para quem busca eficiência. É necessário não somente a ação reativa em caso de problema, também é preciso analisar as situações que gerem a degradação dos sistemas, a fim de tomar ações corretivas antes que falhas ocorram, caracterizando uma abordagem proativa.

Existem muitas ferramentas capazes de auxiliar o administrador nessa tarefa, a escolha do Zabbix deu-se pelas características promissoras das funções disponíveis para gerenciamento de redes, e pela licença de uso da ferramenta, a *General Public Licence* versão 2 (GPL v2), que não tem custo financeiro.

Essa pesquisa objetiva-se a aplicar o Zabbix, no auxílio ao administrador, na identificação de eventos e erros que possam levar ao mau funcionamento de redes distribuídas, corrigindo-os antes que o desempenho seja afetado.

1.1 OBJETIVO GERAL

Implantar e avaliar a ferramenta Zabbix no auxílio ao administrador na busca de soluções para evitar ou corrigir problemas encontrados no gerenciamento de ambientes distribuídos.

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos são listados abaixo:

a) estudar o software Zabbix;
b) aplicar o Zabbix no gerenciamento de ambientes distribuídos;
c) simular eventos e erros de funcionamento no ambiente;
d) documentar os resultados alcançados;
e) validar os resultados e avaliar a ferramenta.

1.3 JUSTIFICATIVA

A popularização das comunicações permitiu as redes de computadores tornarem-se comuns. As pessoas e empresas estão cada vez mais conectadas, essa difusão do acesso e amplo consumo de serviços que ela proporciona são os principais responsáveis por uma nova realidade nas relações profissionais e pessoais (FOROUZAN; MOSHARRAF, 2012).

As empresas, buscando melhores resultados no domínio de suas atividades, adotaram a tecnologia e como consequência obtiveram aumento de agilidade na tomada de decisão, melhorando processos e permitindo um crescimento no desempenho dos colaboradores (OLIVEIRA; ABDALA, 2003).

As organizações na era da informação passam a basear seu modelo de operação em sua infraestrutura de tecnologia (WEILL; ROSS, 2010).

A quantidade de serviços e ativos de rede compartilhados que a infraestrutura de rede da organização tem que suportar cresceu em quantidade e complexidade. Esse aumento deve-se ao fato de que compartilhando esses ativos, a empresa consegue reduzir custos.

Como consequência, a capacidade de geri-los de forma organizada torna-se uma dificuldade a mais para o administrador de redes. Ele trabalha para detectar

e corrigir problemas que tornam a comunicação ineficiente ou impossível e para eliminar as condições que podem produzir o problema novamente.

A necessidade de controle sobre os processos e recursos de rede é muito grande, e difícil de ser feita sem o auxílio de um sistema de gerenciamento de rede.

O software de gerenciamento de rede permitirá ao administrador obter o status dos dispositivos e serviços, gerando estatísticas do seu funcionamento. Seu objetivo principal é alertar o administrador e auxiliá-lo a tomar decisões e contramedidas para a solução dos problemas, caso ocorram (STALLINGS, 2005).

A principal vantagem desta pesquisa é demonstrar os benefícios que o administrador obterá com o uso do sistema de gerenciamento de redes Zabbix, ao monitorar ativos e serviços de rede, além do uso de relatórios gráficos que permitem realizar análises de tendências de comportamento possibilitando ações proativas na detecção de incidentes e resolução de problemas em uma infraestrutura de Tecnologia da Informação (TI).

A escolha do Zabbix deu-se pelo fato de ele suportar uma ampla gama de funções de monitoramento, de checagens simples de disponibilidade dos ativos e dos serviços de rede, a agentes para as mais diversas arquiteturas e protocolos. Vários tipos de notificação como e-mail, mensagens instantâneas e via *Short Message Service* (SMS), módulos para monitoramento de ambientes distribuídos e flexibilidade na configuração.

E por ser um software livre e disponível para download sem custos, incentiva a inclusão da sociedade no seu uso e a difusão de conhecimentos eliminando monopólios sobre a informação.

1.4 ESTRUTURA DO TRABALHO

O projeto é formado por cinco capítulos que apresentam as etapas realizadas. O primeiro apresenta a introdução ao assunto abordado, o objetivo geral, os objetivos específicos que se procurou alcançar e a justificativa para a realização do projeto.

No segundo capítulo são expostos conceitos relacionados com gerencia de redes, por que o administrador precisa gerencia-la e a necessidade de uso de softwares para fazê-lo. Também são descritas as áreas funcionais da gerência segundo a *International Standards Organization* (ISO), de acordo com a ação

aplicada e por fim os componentes da arquitetura de gerenciamento *Transmission Control Protocol/Internet Protocol* (TCP/IP), para entender o modelo adotado pelo Zabbix.

Já no terceiro capítulo são descritas algumas ferramentas de gerência de rede, das mais simples até as mais complexas, para compreender como cada uma é utilizada na resolução de problemas. Uma descrição mais detalhada do Zabbix é apresentada, expondo suas funcionalidades.

O quarto apresenta trabalhos pertinentes na área desse projeto e seus resultados obtidos e no quinto e último capítulo, são demonstrados o trabalho desenvolvido, os testes realizados e os resultados obtidos.

Finalizando, tem-se a conclusão, onde são expostas as considerações finais desta pesquisa e a apresentação de trabalhos futuros.

2 GERÊNCIA DE REDES

As redes de computadores surgem e passam a se popularizar graças ao compartilhamento de periféricos e informações, gerando redução de custos e aumento de produtividade (FARIAS, 2006).

Elas passam a influenciar a vida pessoal e profissional. Decisões corporativas precisam ser tomadas cada vez mais rápido, e quem as toma, precisa ter cada vez mais informações confiáveis para justificar as escolhas (FOROUZAN, 2006).

Tornaram-se estruturas de comunicação extremamente complexas. O elevado número de serviços consumidos pelos usuários obriga o desenvolvimento de soluções para controlar esse nível de utilização (COSTA, 2008).

Administrá-las é uma tarefa difícil pela quantidade de componentes e fabricantes, e também pela possibilidade de os equipamentos serem incompatíveis. Quanto mais itens estiverem envolvidos, maiores serão as probabilidades de problemas ocorrerem e mais difícil será mantê-las organizadas (COMER, 2007).

O controle sobre esses componentes é realizado por meio da gerência, pela obtenção de informações de desempenho e disponibilidade dos elementos que compõe a rede. O conhecimento do comportamento desses elementos permite que sejam tomadas ações na resolução dos problemas (COSTA, 2008).

O termo gerenciamento engloba os aspectos de configuração, controle e relatório, úteis para entender o funcionamento da rede. Esse entendimento serve de auxílio na tomada de decisão durante uma manutenção (FARREL, 2005).

Administrador é o responsável pela saúde da rede. Ele controla e monitora o hardware e o software procurando detectar e corrigir problemas com a intenção de evitar qualquer ineficiência e eliminar condições que possam gerar problemas. Mesmo o hardware e o software de rede sendo capazes de detectar e corrigir falhas de transmissões, à intervenção do administrador é necessária para evitar o baixo desempenho do conjunto (COMER, 2007).

Até mesmo os dispositivos mais simples, precisam ser gerenciados assim que são conectados a rede elétrica (FARREL, 2005).

Além do controle sobre os ativos de rede, itens de hardware e serviços, a gerência também está envolvida com os aspectos de configuração, desempenho e

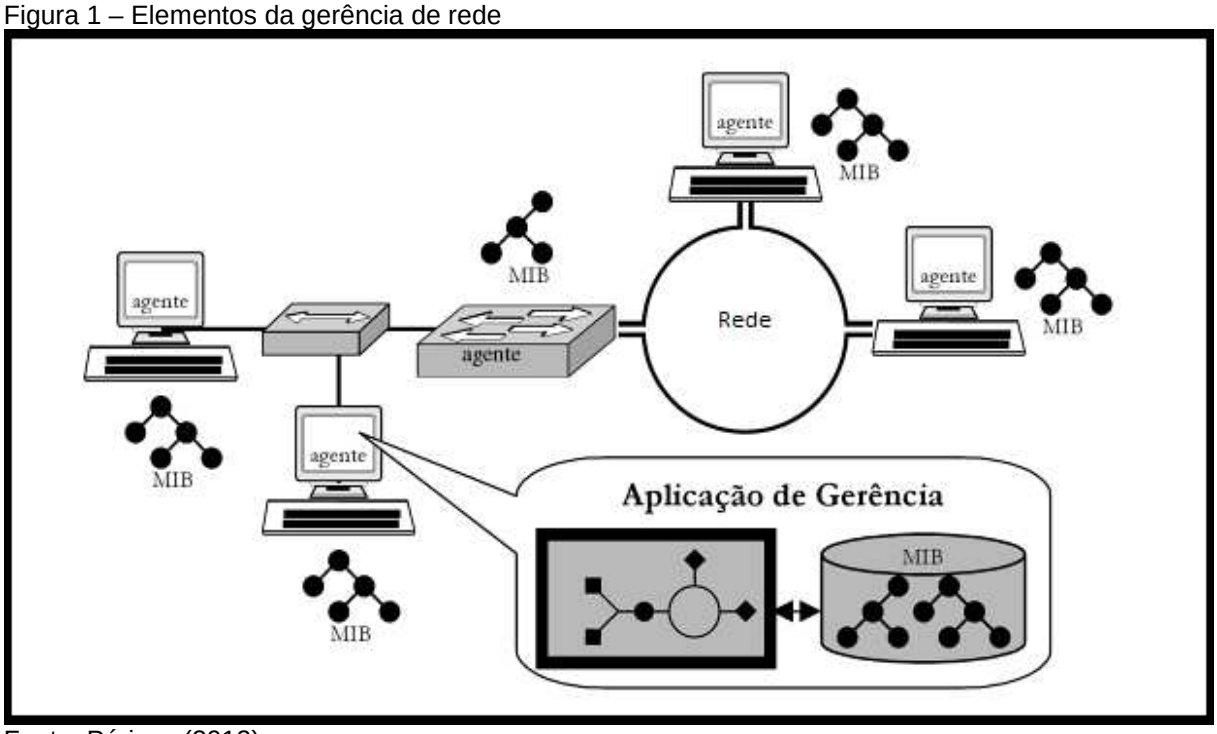
segurança. Praticamente todos os dispositivos de rede ou Internet possuem suporte de gerenciamento (COSTA, 2008).

Normalmente esses dispositivos exigem um grande número de configurações, mas na prática as configurações padrão é que são adotadas, sendo feito somente alguns ajustes de forma a melhorar o funcionamento dessas entidades (FARREL, 2005).

Somente pelo esforço humano é quase impossível a organização e o gerenciamento do ambiente formado pelas redes de uma corporação. A descentralização dos serviços, os ambientes distribuídos e a dispersão dos recursos contribuem para a complexidade, por isso o emprego de ferramentas que automatizem os procedimentos de monitoramento e o controle sobre os itens, tornam-se essenciais (STALLINGS, 2005).

Na figura 1 é possível observar uma rede pequena e os elementos de gerência que a compõem.

Figura 1 – Elementos da gerência de rede



Fonte: Péricas (2012).

Os softwares de gerência auxiliam o administrador a descobrir quando há um problema e isolar suas causas. Fazem isso interrogando os dispositivos e serviços, verificando seu estado para então gerar estatísticas sobre seu funcionamento (COMER, 2007).

Levam em consideração o conhecimento dos possíveis estados de comportamentos que os dispositivos possam apresentar, comparando-os com a informação do estado atual (STALLINGS, 2005).

Eles irão monitorar o funcionamento e notificar o gerente sempre que houver algum evento incomum, fazendo consultas nos ativos e nos serviços de rede para determinar se estão operando corretamente (CASAD; WILLSEY, 1999).

Esses softwares são normalmente conjuntos de ferramentas integradas, que olham a rede como uma arquitetura unificada, permitindo a adição de novos componentes a qualquer instante para ampliar a capacidade de monitoramento (STALLINGS, 2005).

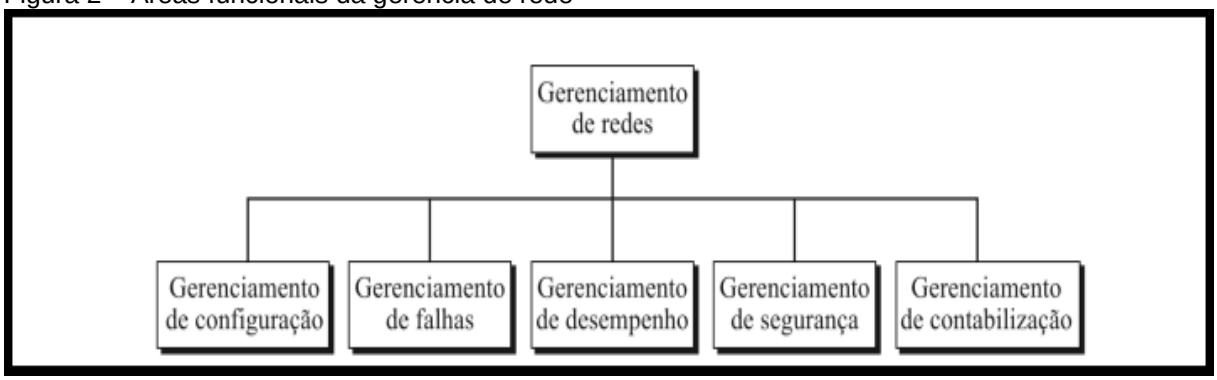
A gerência utiliza protocolos implementados na camada de aplicação do protocolo TCP/IP, para controlar e monitorar os dispositivos e serviços. Quando há a necessidade de interação, o software de gerência estabelece comunicação via protocolos normais como o TCP e o *User Datagram Protocol* (UDP), seguindo o modelo cliente-servidor, fazendo requisições e obtendo respostas (COMER, 2007).

A ISO separou a gerência de redes em áreas de acordo com sua aplicação.

2.1 ÁREAS FUNCIONAIS DA GERÊNCIA DE REDES

A ISO dividiu o gerenciamento de redes em cinco áreas funcionais, baseada nas necessidades dos usuários com intenção de descrever de maneira útil as exigências de qualquer sistema gerenciável de redes. São elas: gerência de desempenho, falhas, configuração, contabilização e segurança, conforme a figura 2:

Figura 2 – Áreas funcionais da gerência de rede



Fonte: Forouzan e Mosharraf (2012).

2.1.1 Gerência de desempenho

O objetivo dela é monitorar e qualificar o desempenho dos componentes de rede sejam eles *hosts*[1], *switches*[2], roteadores[3] ou algo mais abstrato como um trajeto na rede (KUROSE; ROSS, 2006).

Serão mensuradas as capacidades da rede, desde tempo de resposta a tráfego, sempre visando que ela seja a mais eficiente possível (FOROUZAN; MOSHARRAF, 2012).

É necessário o monitoramento de muitos recursos para obter todas as informações necessárias que determinem o nível operacional da rede. Quando analisadas elas permitem ao administrador reconhecer possíveis indícios de degradações. As estatísticas geradas ajudam no planejamento de ações e na tomada da decisão apropriada sobre investimentos (TEIXEIRA JUNIOR et al, 1999).

Pode-se dividir o gerenciamento de desempenho em duas categorias: monitoramento, que é o acompanhamento das atividades e o controle onde são feitos os ajustes para melhorar o funcionamento (STALLINGS, 2005).

Com a implementação de softwares de monitoramento de rede, os administradores de sistema coletam uma quantidade suficiente de dados e geram relatórios periodicamente, o que irá ajudá-los a realizar processos de gestão de forma clara e fácil (KUNDU; LAVLU, 2009, tradução nossa).

2.1.2 Gerência de falhas

Por meio dela é que se mantém uma rede operacional, verificado, componente por componente para quando ocorrer uma falha se saiba exatamente onde ela ocorreu, seja feito o isolamento e o reparo para que a rede retorne ao estado inicial (TEIXEIRA JUNIOR et al, 1999).

Ela irá tratar da detecção e reação às falhas encontradas. A diferença para o gerenciamento de desempenho é o tratamento imediato do problema (KUROSE; ROSS, 2006).

---

[1] Qualquer computador ou máquina conectado a uma rede (SAWAYA, 1999).
[2] Dispositivo usado para conexão física entre dois equipamentos de redes (SAWAYA, 1999).
[3] É um dispositivo que encaminha pacotes de dados entre redes de computadores (VELLOSO, 2014).

Cada componente terá o funcionamento acompanhado, sempre buscando por quaisquer condições anormais que requeiram a tomada de uma ação (STALLINGS, 2005).

Essas ações poderão ser reativas, quando tomadas após as falhas terem sido detectadas e isoladas, ou proativas quando se age preventivamente, antes do problema ocorrer (FOROUZAN; MOSHARRAF, 2012).

2.1.3 Gerência de configuração

Ela vai permitir ao administrador saber quais as configurações dos hardwares e softwares da rede (KUROSE; ROSS, 2006).

Os componentes físicos e os sistemas lógicos podem ser configurados para diferentes aplicações. Um dispositivo poderá funcionar de forma diferente dependendo da situação (STALLINGS, 2005).

Grandes redes são compostas por muitos dispositivos, e apresentam uma configuração quando iniciadas. Com o tempo itens na rede são alterados, computadores substituídos, softwares são atualizados. A gerência de configuração conhecerá o relacionamento entre essas entidades, por meio do registro de seus estados via documentação (FOROUZAN; MOSHARRAF, 2012).

2.1.4 Gerência de contabilização

Nela é feito o controle de acesso aos recursos da rede por usuário e dispositivos. Esse controle se dá por meio de cotas de utilização e cobrança por uso (KUROSE; ROSS, 2006).

O administrador poderá acompanhar também o consumo feito pelo grupo de usuários (TEIXEIRA JUNIOR et al, 1999).

Essa contabilização é feita para impedir que usuários monopolizem recursos ou usem o sistema de forma ineficiente, além de permitir o planejamento dos recursos pela demanda de uso (FOROUZAN, 2006).

2.1.5 Gerência de segurança

Ela fará o controle e monitoramento de acessos à rede. Fornecendo mecanismos para a proteção das informações internas (TEIXEIRA JUNIOR et al, 1999).

Também fará o controle de acesso aos recursos de acordo com a política de segurança definida (FOROUZAN; MOSHARRAF, 2012).

Arquivos de *logs*[4] também são importantes para essa gerência, portanto sua coleta, análise e armazenamento fazem parte do escopo (TEIXEIRA JUNIOR et al, 1999).

2.2 ARQUITETURA DE GERENCIAMENTO TCP/IP

A arquitetura do gerenciamento de redes TCP/IP é composta pelos seguintes itens: Gerente ou entidade gerenciadora, agente ou entidade gerenciada, base de informações gerenciais e o protocolo de gerenciamento.

A entidade gerenciadora é responsável pela aquisição e análise e processamento das informações de rede, onde o administrador interage com os dispositivos gerenciados tomando ações, caso necessário. Essas ações são baseadas nas informações apresentadas (KUROSE; ROSS, 2006).

Ela pode ser um dispositivo ou um software instalado em uma estação, como o Servidor Zabbix.

Podem ainda agregar inteligência, por meio de relatórios de tendências e atuando em tarefas mais complexas (GURGEL et al, 2015).

A entidade gerenciada pode ser qualquer equipamento na rede como computadores, roteadores, *switches* e impressoras. Nessas entidades há um processo, o agente, que é utilizado pelo gerente para comunicação e execução de ações (KUROSE; ROSS, 2006).

O agente permitirá acesso e proverá informações sobre o estado atual do equipamento (BRANCO et al, 2015).

Os dispositivos gerenciados dispõem de uma base de informações de gerenciamento associada aos objetos gerenciados como tráfego de rede e número

[4] Registros de atividades gerados por programas de computador (SAWAYA, 1999).

de pacotes transferidos para uma interface de rede. São esses os valores utilizados pelo gerente, de forma a gerar os relatórios e ajustes caso necessário (KUROSE; ROSS, 2006).

Os objetos são representados por variáveis de dados onde estão armazenadas as informações utilizadas pela entidade gerenciadora (STALLINGS, 2005).

Por fim, há o protocolo de gerenciamento que será executado entre o gerente e o agente, sendo utilizado pelo gerente para a execução de ações nos agentes, ou pelos agentes para informar em caso de falha de algum dos componentes. Ele não gerencia a rede, mas oferece subsídios para que o administrador o faça (KUROSE; ROSS, 2006).

O protocolo vai definir como será feita a troca de informações e quais dados serão trocados entre agente e gerente (COMER, 2006).

Para gerenciamento de redes TCP/IP, o protocolo padrão é o *Simple Network Management Protocol* (SNMP), visto a seguir. Atualmente encontra-se na versão 3. Ele define o formato das mensagens e o conjunto das operações realizadas.

2.2.1 Protocolo SNMP

O SNMP é um protocolo cliente-servidor, operando na camada de aplicação. Ele foi projetado inicialmente para ser independente de protocolos de transporte, sendo possível o envio e o recebimento de informações de gerenciamento (ARNETT, 1997).

A independência de protocolo de transporte o permite operar em vários ambientes de rede. Sua especificação inclui além do protocolo, um banco de dados com as informações de gerenciamento (STALLINGS, 2005).

O protocolo define como o gerente se comunica com o agente, isto é, o formato das requisições bem como o formato das respostas (COMER, 2007).

A coleta das informações poderá ocorrer de duas formas: *polling* e interrupção ou *trap*.

No *polling* o gerente é quem faz as consultas no agente. A pontualidade das informações pode ser afetada pelos intervalos de coleta, onde em caso de intervalos muito curtos pode haver excesso de tráfego, ou em caso de intervalos

longos a notificação de eventos fique comprometida. Via interrupção ou *trap* é a entidade gerenciada que notifica o gerente (ARNETT, 1999).

O modelo SNMP, segue a especificação TCP/IP, com um gerente ou estação de monitoramento, que recupera e analisa as informações de gerenciamento (STALLINGS, 2005).

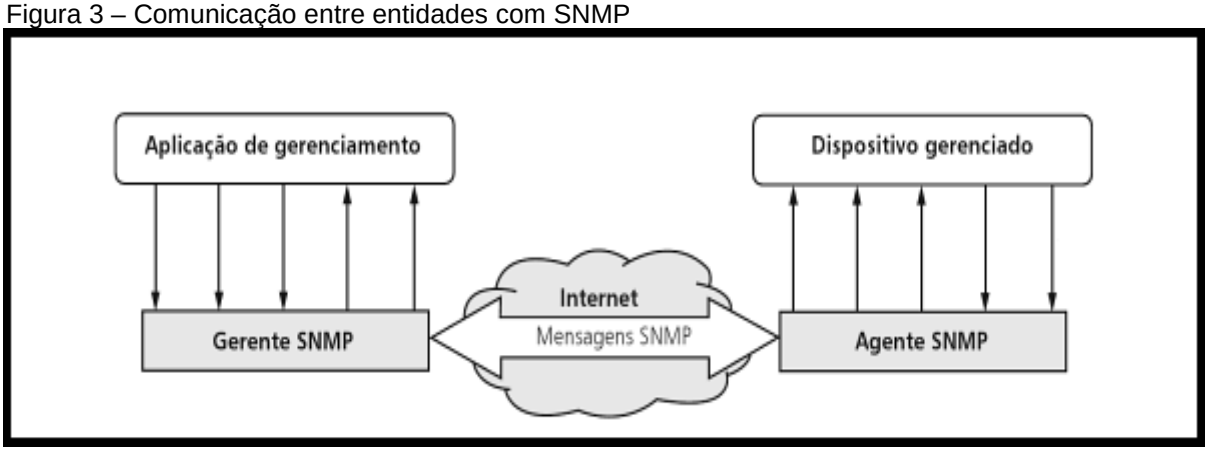
A estação gerenciada ou agente que disponibiliza informações em seu banco de dados ao gerente (FOROUZAN; FEGAN, 2010).

As informações, chamadas de *Management Information Base* (MIB), que são os objetos gerenciados (STALLINGS, 2005).

E o protocolo usado para a comunicação entre gerente e agente. O protocolo SNMP mantém sua estrutura geral parecida desde as primeiras versões (COMER, 2006).

A figura 3 apresenta o fluxo de comunicação com SNMP entre duas entidades da gerência de redes.

Figura 3 – Comunicação entre entidades com SNMP



Fonte: Péricas (2012).

As versões iniciais oferecem segurança simples por meio do uso de comunidades, onde os softwares enviam e recebem informações somente para comunidades específicas, os agentes são configurados para receber e enviar mensagens somente para os gerentes. A partir da versão 3 (SNMPv3), a segurança aumentou e implementou-se a autenticação e controle de acesso (CASAD, 1999).

3 FERRAMENTAS DE GERENCIAMENTO

Existem várias ferramentas que podem ser usadas para gerenciar redes, apresentando maior ou menor grau de complexidade e abrangência na análise de equipamentos ou serviços ativos, por meio da investigação dos protocolos de comunicação e de gerência. Essas ferramentas descritas a seguir, aumentam a eficiência na detecção e resolução de problemas.

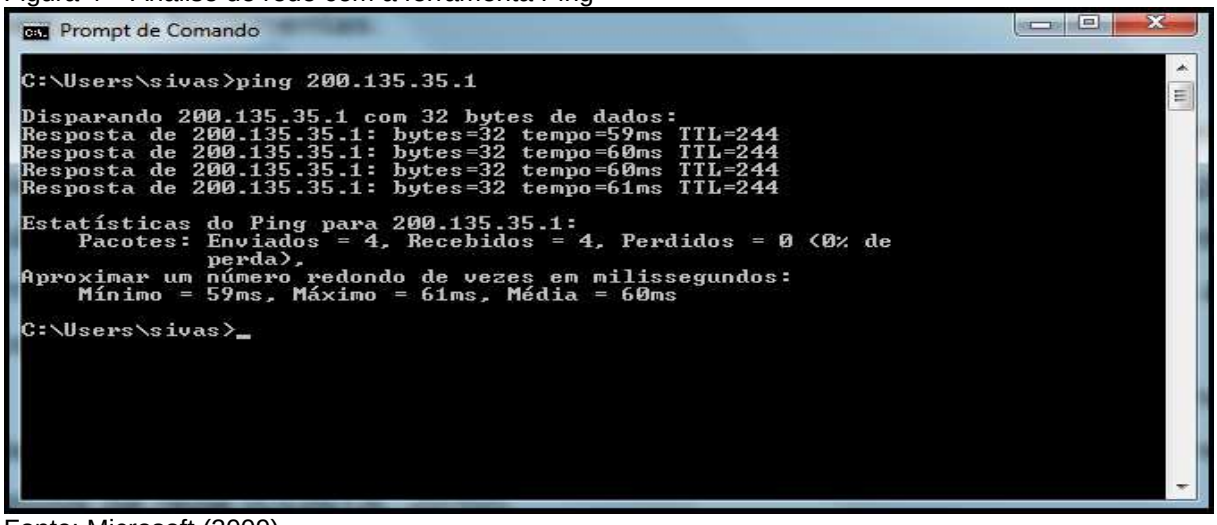
3.1 FERRAMENTAS DE REDE

Ao se analisar os problemas de conexão em rede, é necessário que se faça uma avaliação das causas e por meio delas que se busquem possíveis problemas relacionados (TEIXEIRA JUNIOR et al, 1999).

São ferramentas ou comandos presentes nos sistemas operacionais utilizados para análise das configurações de rede e conectividade. São exemplos as seguintes ferramentas:

O *Packet Internet Groper* (PING) é a ferramenta mais simples desenvolvida para o TCP/IP. Pode tanto validar configurações como diagnosticar se há conectividade entre os computadores por meio do envio de mensagens de eco do *Internet Control Message Protocol* (ICMP[5]). A mensagem de confirmação, figura 4 ainda exibe os tempos de ida e volta (ROSS, 2008).

Figura 4 – Análise de rede com a ferramenta Ping



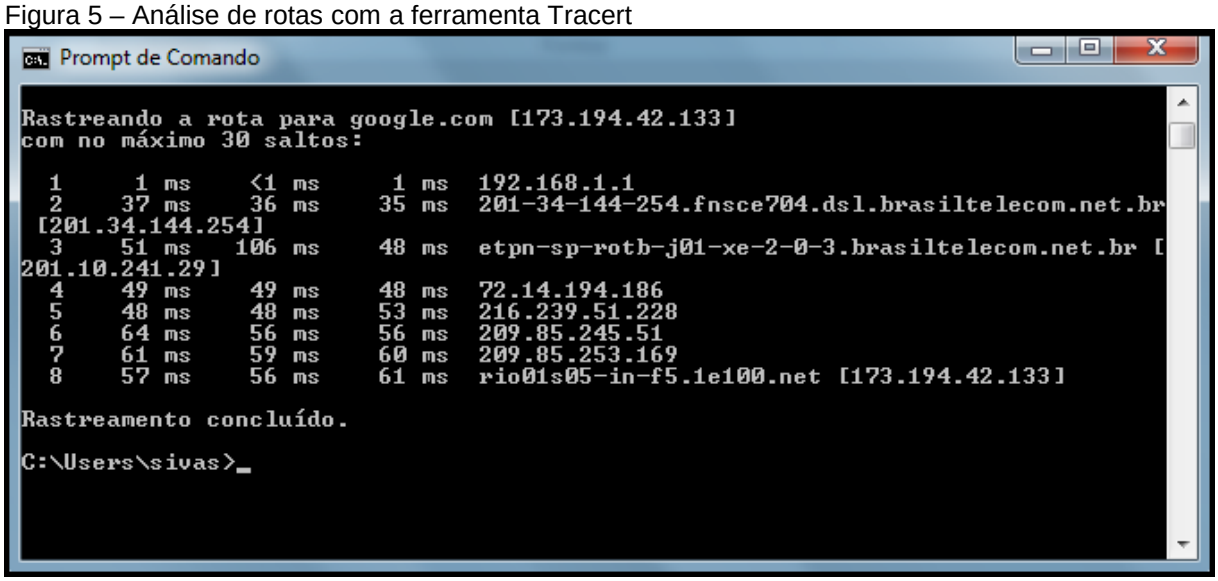
Fonte: Microsoft (2009).

[5] Protocolo utilizado para diagnósticos em redes TCP/IP (ALVES, 2008).

Está disponível em todos os sistemas operacionais modernos, por meio dela pode-se descobrir se há indisponibilidade do destino em caso de não haver resposta da solicitação ou alguma sobrecarga caso o tempo de resposta seja elevado (COSTA, 2010).

O *Traceroute/Tracert* é a ferramenta que permite descobrir a rota dos pacotes em tempo real, da origem até seu destino. Essas rotas são decididas pelos roteadores, sempre pensando no melhor caminho a seguir. Os pacotes armazenam um valor com seu tempo de vida, figura 5, sendo decrementado em cada roteador que passe para evitar que trafegue indefinidamente (COSTA, 2010).

Figura 5 – Análise de rotas com a ferramenta Tracert



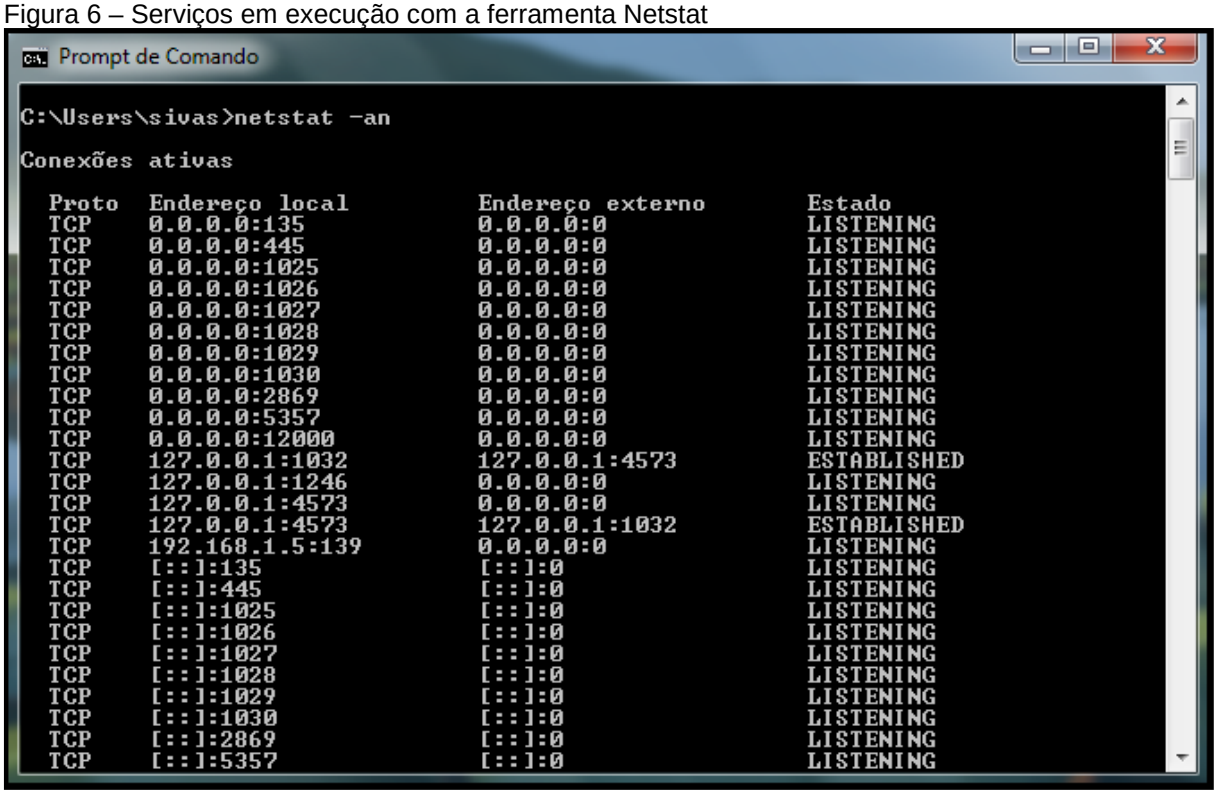
Fonte: Microsoft (2009).

O *Network Statistic* (Netstat) que é utilizado para verificar informações de rede associadas a processos em execução, como protocolos e portas de comunicação (COSTA, 2010).

Também é possível obter estatísticas de envio e recebimento de pacotes e informações de rotas (FARIAS, 2006).

Na figura 6 observam-se os serviços de rede sendo executados pelo sistema operacional e listados pelo comando *Netstat*.

Figura 6 – Serviços em execução com a ferramenta Netstat



Fonte: Microsoft (2009).

3.2 ANALISADORES DE PROTOCOLO

Os analisadores de protocolos servem para inspecionar os pacotes de um protocolo (TEIXEIRA JUNIOR et al, 1999).

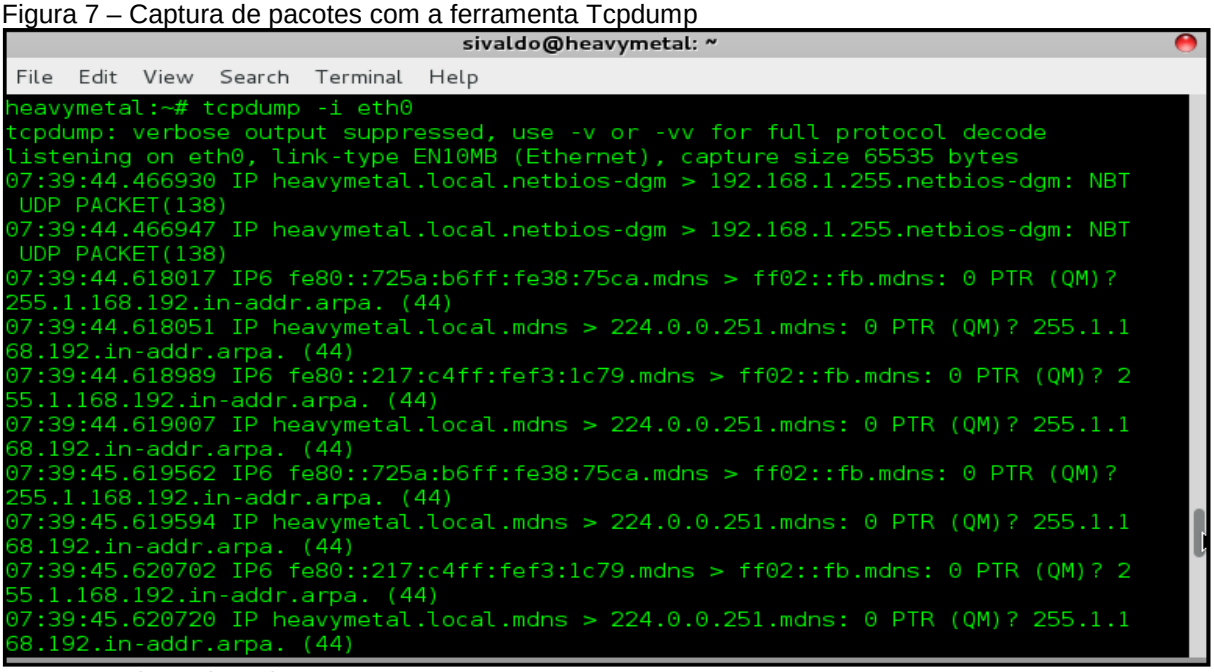
Com eles é possível analisar a fundo os protocolos de rede, verificando o tráfego, de modo que todos os dados transmitidos por ela possam ser interceptados. São exemplos de analisadores de protocolo:

O Tcpdump é uma ferramenta que lê todos os pacotes de uma rede, ou somente os que correspondem a um critério, e fornece informações a respeito (STEVENS; FENNER; RUDOFF, 2005).

Os pacotes poderão ser capturados em qualquer interface de rede ou até em todas para o administrador analisar os resultados (STRANGER; LANE, 2001, tradução nossa).

A tela demonstrando a captura de pacotes no protocolo TCP/IP pode ser vista na figura 7.

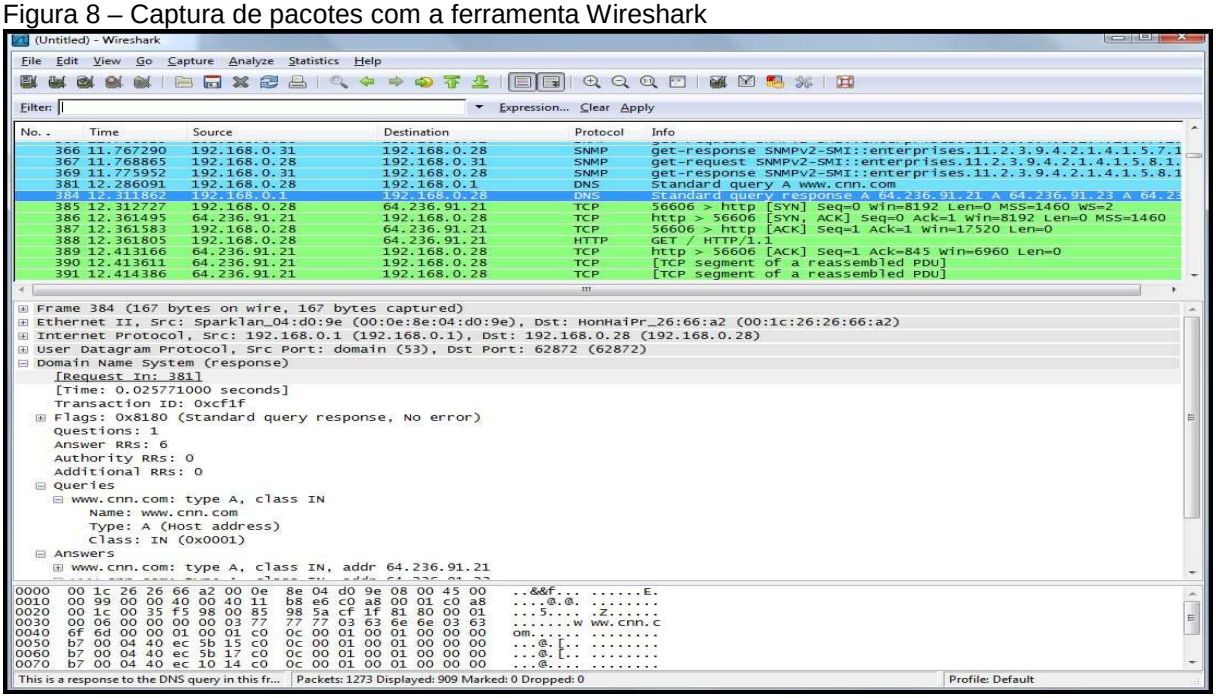
Figura 7 – Captura de pacotes com a ferramenta Tcpdump



Fonte: Tcpdump (2015).

Assim como o *Tcpdump,* o *Wireshark* também captura os pacotes de um protocolo para o administrador analisar, seu diferencial é possuir uma interface gráfica, figura 8, que auxilia no processo de análise. A captura se dá de forma passiva, não havendo qualquer interferência (FOROUZAN; MOSHARRAF, 2012).

Figura 8 – Captura de pacotes com a ferramenta Wireshark



Fonte: Wireshark (2015).

3.3 SISTEMAS DE GERENCIAMENTO DE REDE

São sistemas mais abrangentes, que analisam o desempenho e o status dos componentes da rede em busca de problemas, são compostos por ferramentas para monitoramento e controle, espalhados entre os nós da rede (TEIXEIRA JUNIOR et al, 1999).

São exemplos desses sistemas: Nagios, Zenoss, OpenNMS, Zabbix, entre outras.

O software Nagios, foi criado e ainda é mantido por Ethan Galstad. Sua capacidade de administrar infraestruturas de rede é comparável a sistemas comerciais, podendo ser empregado para gerenciamento de ambientes de grande e pequeno porte, seja monitorando servidores ou arquivos de configuração, além de equipamentos com suporte ao protocolo SNMP (PITANGA, 2008).

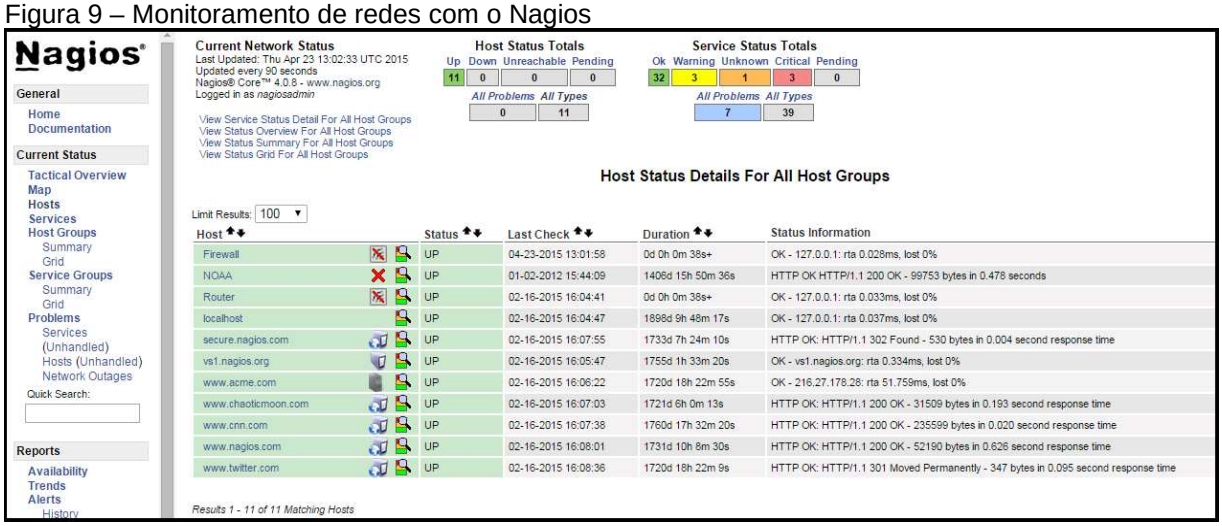
Ele é um software livre para monitoramento de redes, com uma interface Web para gerenciamento. Pode monitorar desde servidores, dispositivos de rede, aplicações ou serviços com um endereço de rede, acessado via TCP/IP. Pode ser configurado por meio de *firewalls*[6], túneis de rede privada virtual e via Internet. Ele pode monitorar uma variedade de propriedades dos ativos de rede, desde processador, memória e disco ao estado de aplicações, arquivos e banco de dados, usando uma variedade de protocolos. Recebe avisos via SNMP e permite a criação de checagens personalizadas em uma variedade de linguagens de programação como C, Perl, Bash. Ainda pode ser configurado como um sistema de monitoramento robusto com redundância a falhas e capacidade recuperação a desastres (TURNBULL, 2006, tradução nossa).

Sua capacidade modular e abordagem simples para o monitoramento o tornam muito fácil de trabalhar e altamente escalável. Além disso, por ser um software livre sua licença permite que seja customizado para atender às necessidades específicas (JOSEPHSEN, 2007, tradução nossa).

A interface do Nagios, com os resultados do monitoramento é vista a seguir na figura 9.

[6] É uma barreira de contenção que ajuda a bloquear o acesso de conteúdo malicioso (VELLOSO, 2014).

Figura 9 – Monitoramento de redes com o Nagios



Fonte: Nagios (2015).

Zenoss é um sistema de monitoramento e solução de gestão empresarial de código livre, que fornece um único ponto de acesso, baseado na Web para configurar, gerenciar, monitorar e informar sobre os ativos de TI. É uma aplicação baseada no Linux escrita na linguagem Python, sua interface permite o gerenciamento dos dispositivos, monitorar desempenho, administrar eventos e alertas além da geração de relatórios (BADGER, 2008, tradução nossa).

Na figura 10 podem ser vistos alguns itens monitorados na tela do Zenoss.

Figura 10 – Itens monitorados com Zenoss

Fonte: Zenoss (2015).

OpenNMS é uma aplicação de gerenciamento de redes construída usando a linguagem de programação Java, sob o modelo software livre. Ele pode executar todas as funções de gerenciamento de rede incluindo, gerenciamento de falhas, gerenciamento de configurações, entre outras. O gerenciamento é feito por meio do controle de falhas e notificações por meio do envio ou recebimento de mensagens. As notificações são usadas para alertar os responsáveis pela rede seja por mensagens de texto via celular a e-mails e criando chamados em sistemas de *help desk*[7]. Os dados de desempenho podem ser coletados via SNMP e Java *Management Extension* (JMX), uma extensão da máquina virtual Java, criada com foco na gerência de redes. A plataforma OpenNMS fornece uma solução de gerenciamento de redes completa e escalável para muitos equipamentos (HACHEY, 2013, tradução nossa).

O portal com os resultados dos serviços sendo monitorados pelo OpenNMS é visto na figura 11.

Figura 11 – Serviços monitorados com OpenNMS

Fonte: Opennms (2015).

3.4 ZABBIX

O Zabbix é um sistema de gerenciamento de redes de código livre e sem custos de licenciamento, que possibilita o monitoramento de vários parâmetros dos

[7] Sistemas que fornecem serviço de atendimento a clientes que procuram por solução de problemas (STATDLOBER, 2006).

ativos de rede, de disponibilidade a desempenho de aplicações e serviços de uma empresa. Foi criado por Alexei Vladishev em 1998 quando trabalhava como administrador de sistemas em um banco, por insatisfação com os sistemas de gerenciamento de rede da época (LIMA, 2014).

Na figura 12 é apresentada a tela com as informações mais relevantes do Zabbix, estado dos serviços e *hosts* monitorados.

Figura 12 – Interface Web do Zabbix

Fonte: Zabbix (2015).

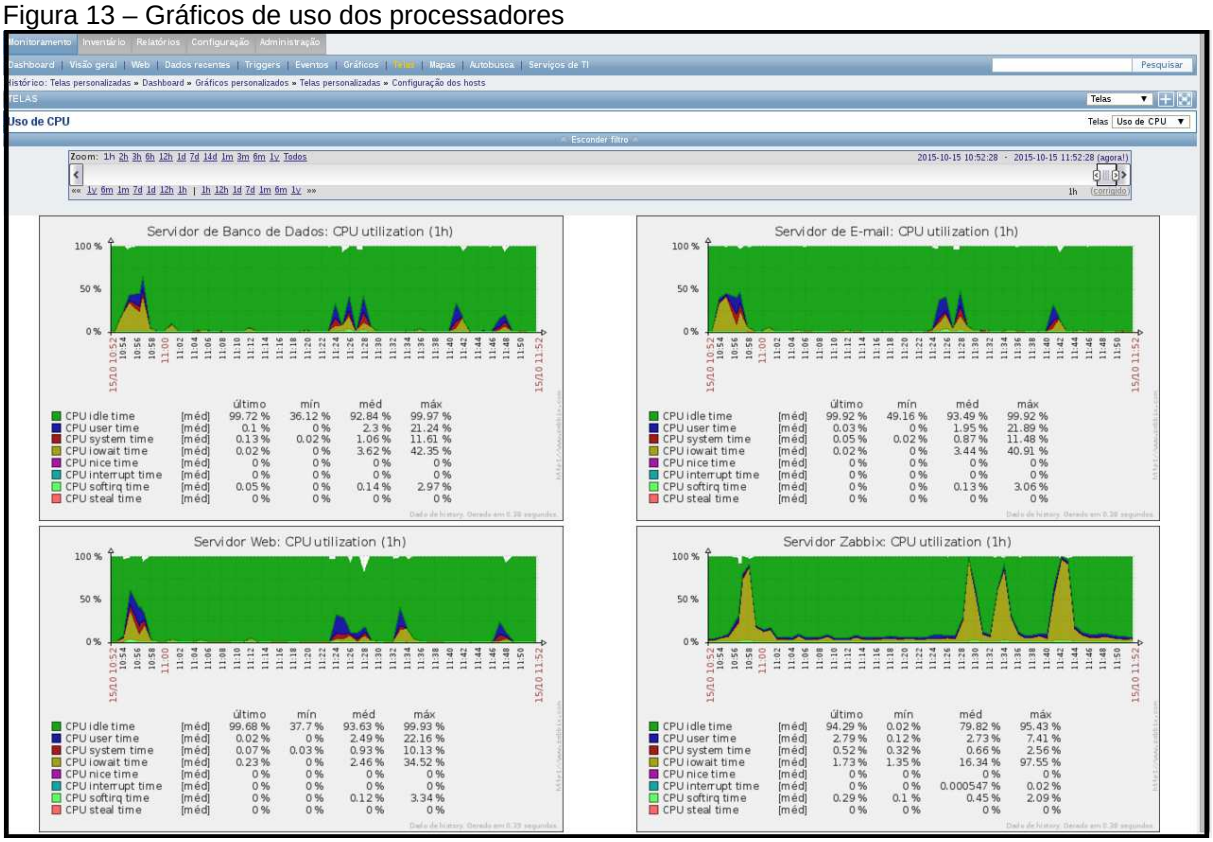
Ele fornece muitas maneiras de monitorar diferentes aspectos da infraestrutura de TI e quase tudo que se conecte a ele. Ele pode ser caracterizado como um sistema de monitoramento distribuído com gerenciamento centralizado. Enquanto muitas instalações tem um banco de dados central, é possível utilizar monitoramento distribuído com nós e *proxys*[8] além de muitas instalações usarem agentes disponíveis nas diversas plataformas (VACCHE; LEE, 2015, tradução nossa).

Possui suporte a autenticação segura com integração a servidores *Lightweight Directory Access Protocol* (LDAP) e controle de permissões flexível, permitindo controlar e auditar as ações no ambiente onde os usuários tem nível de acesso diferenciado. A apresentação dos dados pode ser feita por mapas de rede interativos, gráficos que auxiliam na interpretação das informações captadas e que

[8] Programa de computador que faz a intermediação de uma conexão entre dois pontos (SCHMITT; PERES; LOUREIRO, 2013).

ainda podem ser integrados em telas de apresentação com características específicas, exemplificados na figura 13.

Figura 13 – Gráficos de uso dos processadores



Fonte: Do autor.

O envio de notificações pertinentes em caso de detecção de incidentes, fundamental no gerenciamento de redes, pode ser feito no Zabbix de forma nativa via e-mail, mensagem instantânea via *Jabber*[9], mensagens instantâneas de texto via rede de telefonia celular. Podem-se criar maneiras adicionais para notificação, possibilitando a integração com ferramentas de terceiros (ZABBIX, 2015).

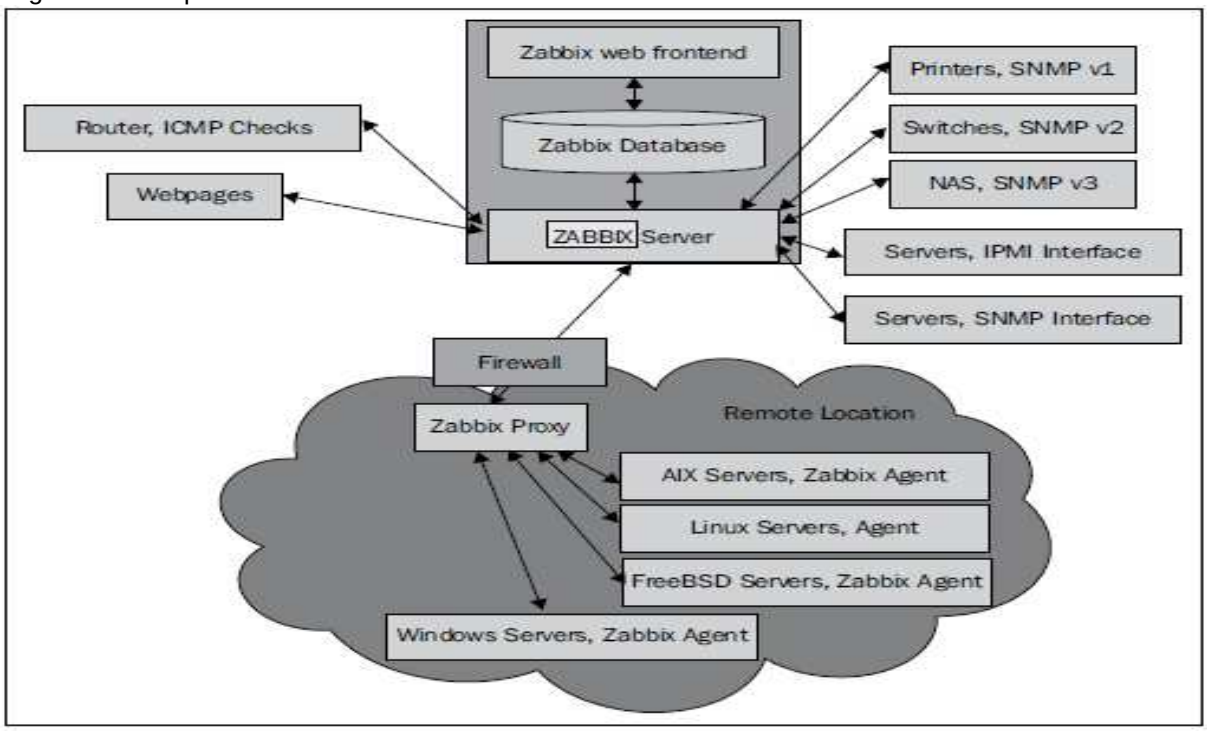
A arquitetura do Zabbix segue o modelo três camadas, com uma camada de aplicação que coleta e processa os dados dos ativos de rede, uma camada de banco de dados que armazena as informações coletadas e uma responsável pela apresentação das informações, representadas na figura 14 (LIMA, 2014).

O servidor é o componente principal que obtém e processa os dados e estatísticas de monitoramento dos agentes e *proxies*. Após o processamento as informações são armazenadas nas bases de dados, alertas podem ser enviados caso alguma condição seja observada. Está disponível para várias plataformas

[9] Protocolo de mensagens instantâneas (HORST; PIRES; DÉO, 2015).

compatíveis com Unix[10] como GNU/Linux, IBM AIX, HP-UX, FreeBSD, OpenBSD, NetBSD e Mac OS X (ZABBIX, 2015).

Figura 14 – Arquitetura Zabbix



Fonte: Vacche e Lee (2015).

Para armazenamento dos dados coletados e configurações são utilizados sistemas de gerenciamento de banco de dados, sendo suportados os bancos livres MySQL/Mariadb, PostgreSQL e os proprietários Oracle e IBM DB2 (HORST; PIRES; DÉO, 2015).

A interface Web é onde a visualização dos dados processados ocorre. É possível personalizar muitos parâmetros, como: idioma, temas, limites na exibição de resultados, reconhecimento de eventos automaticamente, tempo que os dados permanecem armazenados para consulta. As operações de cadastros incluem desde os ativos monitorados até as aplicações e seus respectivos itens monitorados, gatilhos que são ações automatizadas quando ocorre um evento e a geração de gráficos personalizados. A busca de itens adicionados à infraestrutura pode ser configurada para ser realizada automaticamente. A interface foi modelada para utilizar o reaproveitamento das operações já realizadas na forma de *templates*[11],

[10] Sistema operacional (SAWAYA, 1999).
[11] Ambiente estabelecido por modelo (HORST; PIRES; DÉO, 2015)

com herança de propriedades muito semelhante às linguagens de programação orientadas a objetos (ZABBIX, 2015).

Em ambientes menores o servidor Zabbix, o banco de dados e a interface Web podem ser instalados em um único equipamento, para ambientes mais robustos a separação é recomendada.

Dispõem de *proxies* que funcionam como super agentes, ideais para ambientes distribuídos, coletando informações captadas pelos agentes, armazenando em um banco de dados local e então as enviando para os servidores (LIMA, 2014).

A checagem dos serviços e coleta de informações podem ser realizadas mediante o uso de agentes nativos para diversas plataformas como: GNU/Linux, IBM AIX, HP-UX, FreeBSD, OpenBSD, NetBSD, Mac OS X e Microsoft Windows versões desktop e servidor. Há suporte para todas as versões do protocolo SNMP, podendo-se consultar o dispositivo monitorado ou ainda o recebimento de alarmes por meio de operações de *Trap*. Também suporta *Intelligent Platform Management Interface* (IPMI[12]). Agentes *Secure Shell* (SSH[13]) e *Telnet* podem ser usados ainda para o caso de hardware proprietário que não suporte os agentes anteriores. Bancos de dados podem ser monitorados via *Open Database Connectivity* (ODBC[14]), serviços de rede via checagem simples, onde a porta de comunicação do serviço é testada além da monitoração Web, onde procura-se constatar se as informações do site estão no ar, não somente a porta de comunicação (HORST, PIRES; DÉO, 2015).

Há suporte a tecnologia JMX, por meio do *gateway Java*, que garante acesso padronizado às informações na máquina virtual Java e nos servidores de aplicação.

Informações complementares relacionadas com os assuntos abordados são vistas as seguir nos trabalhos correlatos.

[12]É um padrão usado para monitoramento de hardware em servidores. Pode ser usado via TCP/IP para detectar o mau funcionamento de fontes, memórias entre outras funcionalidades (HORST; PIRES; DÉO, 2015).

[13]Protocolo criptografado de conexão e comunicação com um equipamento de rede (STATO FILHO, 2009).

[14]É uma especificação de interface para acesso a dados, provendo funções para conectar e desconectar fontes de dados (SAWAYA, 1999).

4 TRABALHOS CORRELATOS

Esta sessão relaciona alguns trabalhos científicos pertinentes ao tema deste projeto de pesquisa, utilizados em seu desenvolvimento.

4.1 IMPLANTAÇÃO DA FERRAMENTA NAGIOS PARA MONITORAÇÃO DE REDE E ANÁLISE E TRATAMENTO DOS EVENTOS POR MEIO DE SOFTWARES DE APOIO

Trabalho de Conclusão de Curso de Leandro Koehler Cardoso, para obtenção do Grau de Bacharel em Ciência da Computação em 2011, pela Universidade do Extremo Sul Catarinense no estado de Santa Catarina.

O trabalho avaliou o comportamento do Nagios no levantamento de dados em problemas detectados na rede. O estudo foi realizado em um ambiente controlado, simulando eventos e erros para verificar o desempenho dessa rede. As ferramentas propostas no trabalho, Nagios e Cacti, foram avaliadas com o objetivo de saber se são eficazes, no auxílio aos profissionais que desejam utiliza-las como solução de monitoramento.

Chegou-se ao resultado de que ferramentas de monitoramento são cada vez mais necessárias devido à complexidade dos ambientes e eficácia no auxilio a gestão de redes que elas proporcionam.

4.2 SOLUÇÃO DE GERENCIAMENTO DE REDES UTILIZANDO O SISTEMA DE CÓDIGO ABERTO ZABBIX

Monografia de Alisson Andrey Puska, apresentada como requisito parcial para obtenção do grau de especialista em Configuração e Gerenciamento de Servidores e Equipamentos de Redes, do Departamento Acadêmico de Eletrônica da Universidade Tecnológica Federal do Paraná em 2011.

O trabalho demonstrou a utilização da ferramenta de gerenciamento de redes e servidores Zabbix, na montagem uma solução barata e funcional para auxiliar na gerência de uma estrutura de rede utilizando-se software livre. Para isso foi documentada a instalação, configuração e o monitoramento da rede.

Como resultado, observou-se a facilidade da montagem da infraestrutura e a rapidez na obtenção das informações de gerenciamento, mostrando que o Zabbix é uma solução eficaz no monitoramento de redes.

4.3 GERENCIAMENTO DE UMA REDE DE COMPUTADORES EM UM AMBIENTE CORPORATIVO UTILIZANDO O SOFTWARE ZABBIX

Monografia de Leonardo Batista Nunes, apresentada ao curso de Licenciatura em Computação da Universidade Estadual da Paraíba para obtenção do grau em Licenciatura em Computação em 2014.

O trabalho implementou uma solução de gerência em um ambiente corporativo fazendo uso da ferramenta Zabbix, para monitoramento de uma rede de computadores cuja finalidade era monitorar a disponibilidade e o desempenho dos dispositivos presentes na rede permitindo um controle maior e aumentando a qualidade dos serviços oferecidos.

Os resultados obtidos permitiram ao gerente realizar um planejamento para expansão dos serviços de infraestrutura a fim de suprir melhor a demanda dos usuários e melhora no tempo de resolução de problemas dos usuários.

4.4 MONITORIZAÇÃO DE SISTEMAS DE INFORMAÇÃO CRÍTICOS

Dissertação de mestrado integrado de Vladimiro Florival Sousa da Rocha Pinto de Macedo, apresentado ao curso de Engenharia Informática e Computação da Universidade do Porto em Portugal em 2011.

No trabalho é demonstrado um estudo sobre as várias possibilidades de monitoramento de sistemas críticos, apresentando as várias plataformas de monitoramento livres, sendo avaliadas as que melhor correspondiam aos requisitos da instituição.

Como resultado dos testes, o trabalho opta pelo uso do sistema Icinga, devido a extensa base de plugins disponível, o armazenamento de informações, em formato compatível, com outras soluções, interface de operação intuitiva e a organização da equipe de desenvolvimento.

4.5 COMPARAÇÃO DE FERRAMENTAS DE GERENCIAMENTO DE REDES

Trabalho de conclusão de curso de Tomas Lovis Black, apresentado como requisito parcial para obtenção do grau de Especialista no curso de Especialização em Tecnologias, Gerência e Segurança de Redes de computadores na Universidade Federal do Rio Grande do Sul.

Neste trabalho é feita a apresentação e comparação entre ferramentas de monitoramento e gerenciamento de rede baseadas nas características de cada uma, levando-se em conta as vantagens e desvantagens de seu uso, auxiliando na escolha da melhor ferramenta de acordo com as necessidades baseado em parâmetros relevantes como: desempenho, facilidade de utilização e necessidade de recursos.

Como resultado chegou-se a conclusão que o uso combinado de ferramentas seria mais adequado na obtenção de uma solução mais abrangente para o melhor uso dos recursos computacionais.

5 IMPLANTAÇÃO DA FERRAMENTA DE GERÊNCIA DE REDES ZABBIX NO GERENCIAMENTO DE REDES DISTRIBUÍDAS

Este estudo teve como objetivo implantar a ferramenta Zabbix testando e avaliando os resultados de sua eficácia, no auxílio ao administrador de redes, na detecção de erros e situações de degradação no gerenciamento de um ambiente distribuído.

Para isso foi utilizado um dos laboratórios de informática da própria universidade para implantação do ambiente simulado e dos testes relacionados.

Foram monitorados e notificados quaisquer incidentes que ocorreram com os ativos e serviços de rede do ambiente. As informações apuradas com o monitoramento foram utilizadas para geração de relatórios e gráficos de tendências de comportamento, auxiliando a ação proativa do administrador, na resolução de problemas.

A escolha do Zabbix deu-se pela sua ampla gama de funções de monitoramento, de checagens simples de disponibilidade dos ativos e dos serviços de rede, a agentes para as mais diversas arquiteturas e protocolos. Vários tipos de notificação, módulos para monitoramento de ambientes distribuídos e flexibilidade na configuração. E pela licença GPLv2, por não haver cobrança de uso e permitir a participação no projeto.

5.1 METODOLOGIA

A etapa inicial do projeto compreendeu o levantamento bibliográfico em livros, monografias, dissertações e documentação na Internet. Tomando-os como referência houve o embasamento necessário para a elaboração do trabalho de conclusão de curso.

Foram realizados estudos sobre gerência de redes e áreas funcionais para obter conhecimento no foco de aplicação do projeto, as gerências de falhas e desempenho em ambiente distribuído. Também foram estudados a arquitetura de gerenciamento de redes TCP/IP e o protocolo SNMP para compreender o modelo de gerência de redes implementado pelo Zabbix.

O ambiente e os testes aplicados procuraram exemplificar situações cotidianas que ocorrem nas redes de uma organização com filiais. O estudo não

explora todos os possíveis casos encontrados em um ambiente real, em função da quantidade de dados a serem analisados.

Os testes foram direcionados para rotinas de desempenho e detecção de falhas de funcionamento, dos elementos componentes do ambiente, para resolução de problemas. As informações obtidas na infraestrutura pela ferramenta na forma de gráficos de tendências de comportamento são utilizadas para auxílio e planejamento de ações antecipando-se a possíveis problemas.

5.1.1 Modelagem do ambiente do estudo

O ambiente de estudo proposto foi um laboratório de testes composto por um servidor IBM x3500 M4 com um processador Intel Xeon E5-2620 2.00GHz com 12 núcleos, 16GB de *Random Access Memory* (RAM) e um disco de 1TB, com o sistema operacional Ubuntu Desktop 14.04 *Long Term Support* (LTS) como hospedeiro. Foram criadas dez máquinas virtuais utilizando o virtualizador Oracle VM VirtualBox 4.3.30. Nessas máquinas foi instalado o sistema operacional Debian GNU/Linux 8.2. O Zabbix utilizado nos testes do ambiente foi a versão estável mais recente, Zabbix 2.4.

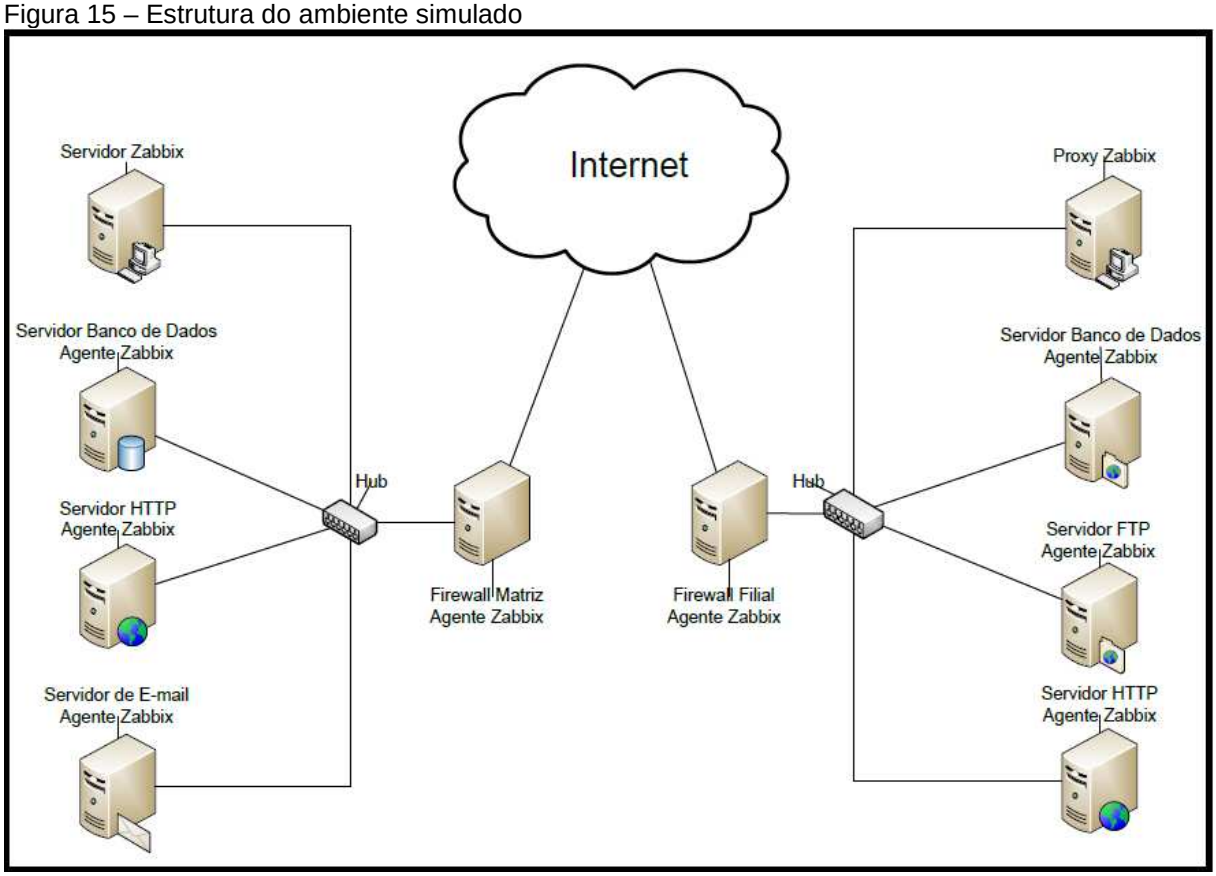
O arranjo das máquinas virtuais permitiu a simulação de um ambiente empresarial, com uma infraestrutura de servidores representando a matriz de uma empresa, provendo um servidor Zabbix para o monitoramento, um de e-mail com os serviços *Post Office Protocol* (POP[15]), *Internet Message Access Protocol* (IMAP[16]) e *Simple Mail Transfer Protocol* (SMTP[17]), um de banco de dados, um Web e um como *firewall* para isolar a rede. Da mesma forma a filial simulada possui uma infraestrutura contando com um servidor de banco de dados, um Web, um de arquivos *File Transfer Protocol* (FTP[18]), um executando o módulo *proxy* do Zabbix além de um *firewall* também para isolar a rede. Elas se comunicam via *firewalls*, por meio de uma *Virtual Private Network* (VPN[19]), simulando equipamentos interligados via Internet. O uso de VPN é necessário porque a informação trafegada entre servidor Zabbix e *proxy* ou agente, não é criptografada. Todos os equipamentos

[15] Protocolo de recebimento de mensagens de *e-mail* (FOROUZAN; MOSHARRAF, 2012).
[16] Protocolo de recebimento de mensagens de *e-mail* com mais recursos (TANEMBAUM, 2003).
[17] Protocolo utilizado para transferir *e-mails* de um cliente para um servidor (COSTA, 2008).
[18] Protocolo de transferência de arquivos (FOROUZAN; MOSHARRAF, 2012).
[19] Rede privada virtual, comunicação criptografada entre dois pontos em uma rede (SAWAYA, 1999).

utilizam o protocolo TCP/IP para comunicação. Todos os servidores virtuais da infraestrutura tem o agente Zabbix instalado.

O ambiente exemplificado na figura 15, proporcionará condições de testes semelhantes a uma infraestrutura de produção com todos os recursos necessários.

Figura 15 – Estrutura do ambiente simulado



Fonte: Do autor.

5.1.2 Ativos de rede e serviços gerenciados

Uma infraestrutura empresarial possui ativos de rede e disponibiliza diversos serviços necessários ao andamento das atividades no ambiente corporativo. Esses ativos e serviços podem não estar todos agrupados em um único local físico. O uso de ferramentas de comunicação como o e-mail e mensageiros instantâneos, o acesso às informações gerenciais, como os sistemas de gestão integrados, e os vários equipamentos que os hospedam nessa estrutura, são essenciais nas rotinas administrativas de uma empresa e suas filiais. A gestão desses itens em um ambiente distribuído dentro de uma corporação pode ser complexa e ineficiente.

Em situações de indisponibilidade, sejam problemas em equipamentos ou falhas nos serviços oferecidos, é comum o setor responsável ser avisado que algo está errado, só então se toma alguma medida para restabelecer o ambiente.

A capacidade de mensurar o consumo dos serviços e planejar melhorias estratégicas também é uma tarefa difícil, mais ainda em ambientes distribuídos ou quando precisa ser realizada manualmente. Tráfego de rede, vazamentos de memória, elevadas cargas de processamento e o descontrole no uso dos sistemas de armazenamento são itens fundamentais capazes de individualmente paralisarem um ambiente se forem mal gerenciados.

5.1.3 Implantação do ambiente

O trabalho foi realizado a partir da instalação do sistema operacional Ubuntu Desktop 14.04 LTS no servidor IBM x3500 M4 como hospedeiro. Nele foi instalado o virtualizador Oracle VM VirtualBox 4.3.30. E para as máquinas virtuais foi instalado o sistema operacional Debian GNU/Linux 8.2. São nelas que a infraestrutura e os módulos do Zabbix foram implementados.

Na instalação da máquina virtual Debian optou-se por um perfil mínimo, onde somente o essencial é instalado, evitando a inclusão de serviços desnecessários. Após a instalação foi executada a atualização do sistema operacional via Internet, utilizando o gerenciador de pacotes da distribuição, para que o sistema base estivesse com a última versão estável dos pacotes disponibilizados.

O recurso de clonagem disponível no Oracle VM VirtualBox foi utilizado para criação das demais máquinas virtuais necessárias baseadas na instalação inicial do Debian. Os passos para a criação do ambiente composto por matriz e filial e dos demais itens são descritos no Apêndice A deste trabalho.

Para a instalação dos módulos do Zabbix foi utilizado o repositório disponibilizado pelo fabricante Zabbix LLC. A instalação via repositório automatiza a resolução de dependências, instalando todos os demais pacotes necessários ao funcionamento do software.

O agente Zabbix foi escolhido como a forma de coleta de dados, pela facilidade que apresenta na obtenção das informações. Ele foi configurado para se

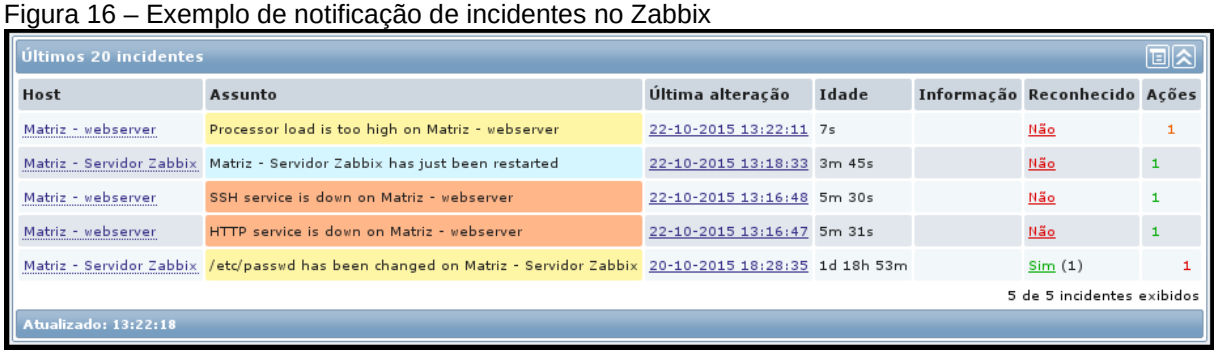
comunicar com o servidor e o *proxy* Zabbix de forma passiva, ou seja, o servidor ou o *proxy* solicitam informações aos agentes. Ele foi instalado em cada máquina monitorada.

O *proxy* Zabbix é uma das peças fundamentais para essa infraestrutura. A quantidade de dispositivos gerenciados em ambientes distribuídos pode ser escalável, podendo ocasionar perdas nas transferências se eles forem monitorados diretamente, devido à quantidade de dados coletados, ou em caso de queda da conexão. Ele armazenará os dados do ambiente remoto antes de enviá-los para o servidor.

O conjunto gerente da infraestrutura composto por servidor Zabbix, sistema de banco de dados e a interface administrativa foi instalado em uma única máquina virtual, por se tratar de um ambiente de testes pequeno e que não demanda alto desempenho. Os passos para a criação da infraestrutura de monitoramento são descritos no Apêndice B deste trabalho.

Na configuração padrão, sempre que um incidente é detectado e permanece ativo, é exibida uma notificação na interface gráfica, alertando e permitindo que seja feito o acompanhamento pelo responsável. Esses alertas apresentam alguns níveis de severidade, desastre, alta, média, atenção, informação e não classificada, de acordo com o grau do problema e representados por cores diferentes configuráveis. A figura 16 a seguir demonstra alguns exemplos de notificações de serviços e itens com problemas. As informações por padrão estão em inglês, mas podem ser personalizadas.

Figura 16 – Exemplo de notificação de incidentes no Zabbix



Fonte: Do autor.

O envio de notificações foi configurado para ser feito por e-mail, para que o administrador fique ciente do incidente, mesmo quando não está acompanhando a

interface do Zabbix. O uso de mensagens de texto via rede de telefonia móvel e *Jabber* não foram implantados por falta de recursos.

As informações de monitoramento, capturadas pelos agentes, são utilizadas para a geração de gráficos sobre o comportamento do ambiente e também servem para o administrador basear suas ações, antecipando-se a possíveis problemas gerindo a rede de forma proativa.

5.2 RESULTADOS OBTIDOS

Nesta etapa, foram realizados os monitoramentos e os testes dos servidores e dos serviços no ambiente.

O software Zabbix demonstrou capacidade no gerenciamento da infraestrutura, não apresentando grande complexidade para configuração. Os resultados obtidos por meio dos testes feitos com a ferramenta foram favoráveis as rotinas de trabalho do administrador.

Nos testes realizados visando detectar gargalos de desempenho ocasionados pelo consumo excessivo de processamento, foi utilizado o software *stress*. Disponível na árvore de pacotes do Debian GNU/Linux, ele executou testes em vários itens do sistema simulando uso excessivo dos componentes.

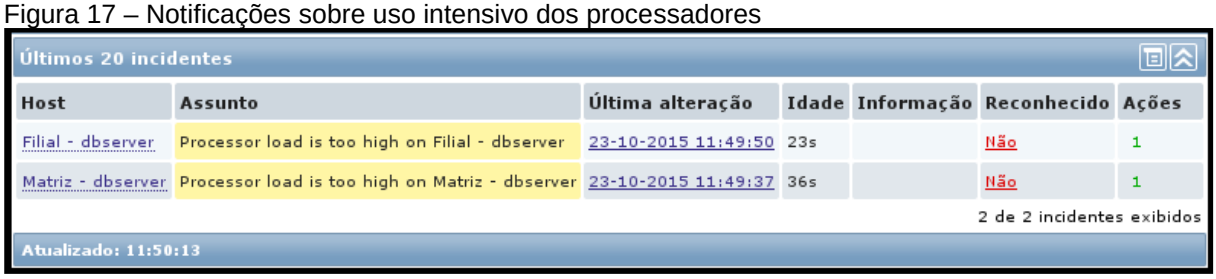
Para a execução dos softwares foi criado um *script* na linguagem do *Shell*[20] Bash, que funciona como gatilho, disparando testes, Apêndice C.

Assim que o uso da *Central Process Unit* (CPU) aumentou o Zabbix fez a detecção e disparou a notificação sobre a ocorrência em ambos os servidores, na tela de incidentes, figura 17.

Os tempos de resposta para o ambiente remoto e o local diferem, pois os dados remotos passam por uma camada extra na infraestrutura, o *proxy Zabbix*. A vantagem de utilizá-lo em relação ao monitoramento direto é que, mesmo se houvesse perda da comunicação com o servidor, às informações ainda estariam armazenadas no banco de dados do *proxy*.

[20] Programa interpretador de instruções no sistema operacional (JARGAS, 2008).

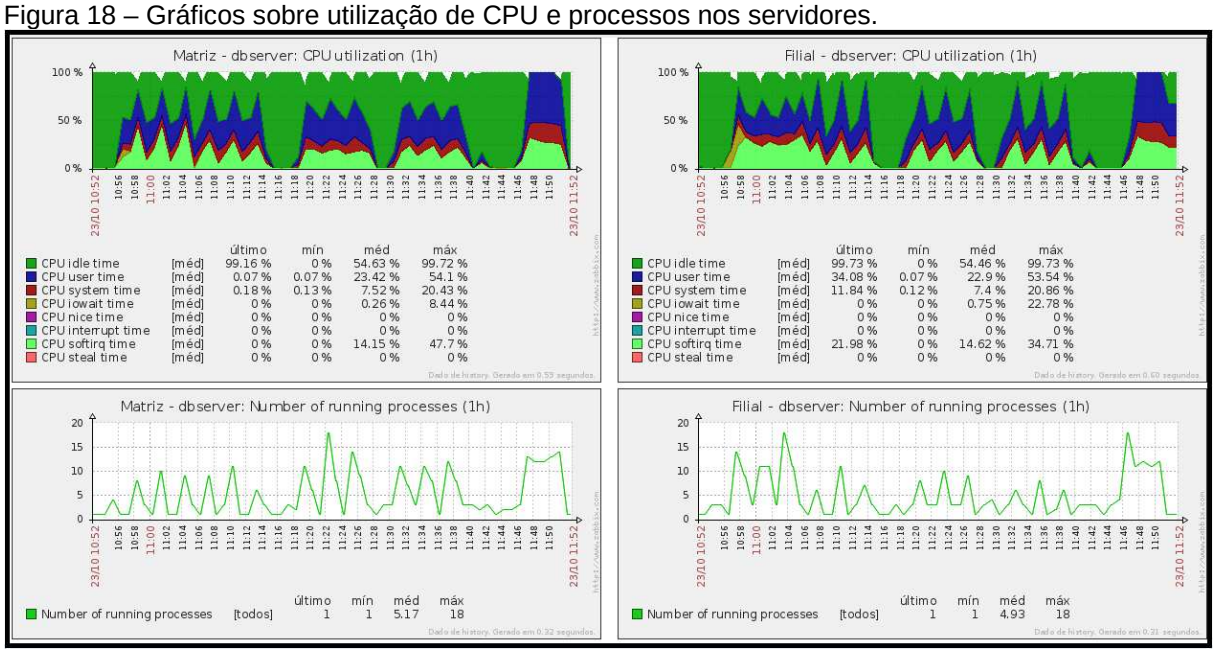
Figura 17 – Notificações sobre uso intensivo dos processadores



Fonte: Do autor.

Analisando os gráficos de utilização de CPU, gerados pelos dados obtidos, é possível detectar que ocorrem picos de processamento nos servidores. Os gráficos de processos executando indicam ainda que durante esses picos de processamento o número de processos em execução aumentava consideravelmente. A figura 18 exemplifica o ocorrido.

Figura 18 – Gráficos sobre utilização de CPU e processos nos servidores.



Fonte: Do autor.

Munido dessas informações o administrador saberá que há processos com problemas, e fará uma investigação mais detalhada nos servidores para solucionar o problema.

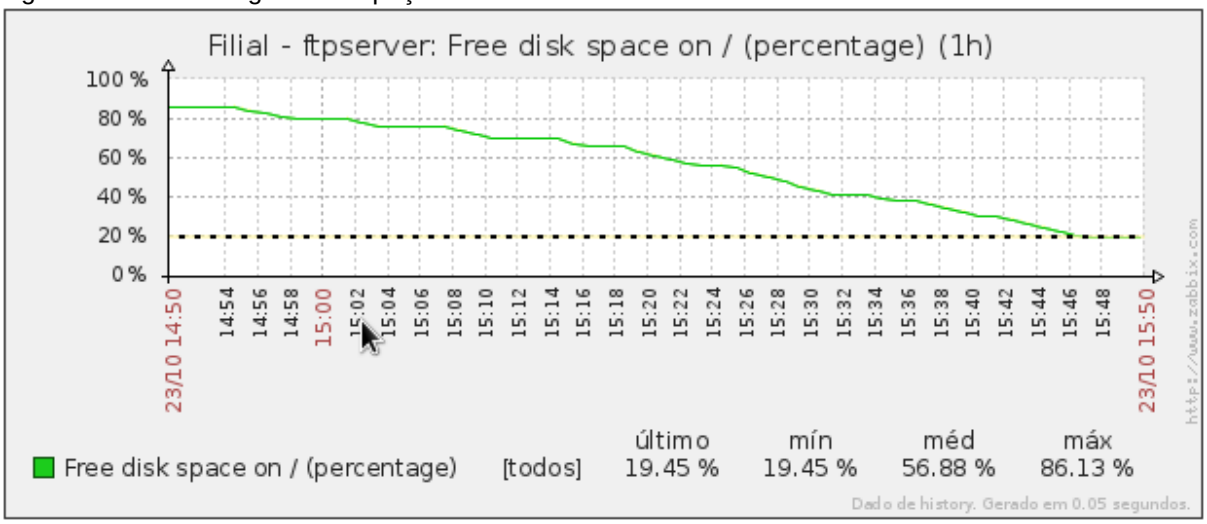
Há situações aonde a degradação do ambiente vai ocorrendo de forma sutil com o passar do tempo, prevê-las é difícil e fundamental para evitar que causem a parada do sistema. A interpretação das informações coletadas pelos agentes por meio dos gráficos gerados é chamada de análise de tendências. O administrador, de posse dessas informações irá se antecipar a ocorrência de falhas.

O teste realizado consistiu em adicionar arquivos ao disco do servidor, via transferência FTP, por um período de tempo até quase saturá-lo. Servidores de arquivos podem apresentar esse problema, principalmente se não são acessados diariamente pelo administrador para que sejam verificados. Por meio do gráfico, foi possível perceber que o espaço livre disponível foi diminuindo com o passar do tempo, mostrado na figura 19. Se o processo continuasse fatalmente ocorreria uma falha no servidor.

Neste caso a ferramenta auxiliou o administrador, obrigando-o a interferir no problema de falta de espaço em disco de forma proativa.

O monitoramento de discos e partições do Zabbix, na configuração padrão ainda disparou uma notificação quando o espaço livre ficou menor que 20% do total. O administrador poderia configurar este valor conforme sua necessidade.

Figura 19 – Porcentagem de espaço livre em disco



Fonte: Do autor.

A mesma análise poderia ser aplicada aos níveis de consumo de memória RAM ou links de Internet.

Normalmente em situações de falta de espaço é comum que um novo disco seja adicionado a estrutura do servidor. Fica para o administrador a tarefa de mapear e configurar esse novo disco, incluindo-o no sistema de gerenciamento de rede. Um diferencial do Zabbix é possuir um processo de descoberta interna automática, que pode ser configurado para descobrir desde discos novos e partições a interfaces de rede. Isso incide positivamente na redução de trabalho do

administrador, modificando o perfil de monitoramento do servidor sem a necessidade de sua interferência.

Existem muitos serviços de rede sendo consumidos pelos usuários nas empresas. A parada desses serviços é um transtorno muito grande para os processos organizacionais.

A checagem simples de portas, para verificar se o serviço está funcionando, é um item de monitoramento muito importante em uma infraestrutura.

Para esse teste, a checagem de portas foi executada em conjunto com a verificação de disponibilidade, coletando o retorno de pacotes ICMP *reply* [21] em um servidor de e-mail executando os serviços POP, IMAP e SMTP. Na simulação a interface de rede do servidor foi desativada. A figura 20, mostra a notificação gerada pela falta de resposta dos serviços.

Na notificação é possível ainda notar o alerta de ausência de respostas para pacotes ICMP, indicação de que o servidor pode não estar ativo ou há algum possível problema de conectividade. A interpretação agregada dessas informações melhora o diagnóstico do problema pelo administrador.

Figura 20 – Notificação de serviços parados no servidor

Fonte: Do autor.

O Zabbix ainda possui um sistema de ações integradas com os gatilhos de eventos. Quando eles ocorrem, as ações podem executar comandos remotos nos servidores, possibilitando reiniciar os serviços, caso o problema tivesse ocorrido somente nos serviços de rede, diminuindo o tempo de resposta para alguns incidentes, e mesmo eles retornando a ferramenta notificaria o administrador sobre a queda.

[21] Resposta para um comando ICMP echo enviado (TANENBAUM, 2003).

Outra variação do teste de checagens simples foi feita em serviços Web. Quase toda organização usa algum tipo de serviço assim. O Zabbix monitorou além da porta do serviço, a disponibilidade e o desempenho do site, simulando a experiência de acesso do usuário. Esse diferencial, chamado de checagem Web, serve para detectar uma falha no site ao invés de um problema no serviço somente.

Na instalação padrão do servidor Web apache2, uma página com conteúdo em *HyperText Markup Language* (HTML) é criada automaticamente no diretório público da aplicação.

A checagem Web consiste em buscar alguma informação diferenciada no endereço monitorado do servidor, podendo ser um texto ou até mesmo a página resultante de uma autenticação no site.

Na notificação do teste vista na figura 21, a página padrão foi removida do diretório e como resultado o alerta não indica problema no serviço, somente no conteúdo ofertado. A ação do administrador é facilitada também nesse caso.

Figura 21 – Notificação de serviços web

Fonte: Do autor.

Outro item que é muito exigido nos servidores é a memória. Alguns serviços acabam utilizando esse recurso ainda mais quando são muito requisitados, é o caso de servidores de aplicação e Web. Há também a chance de existir uma falha, não prevista, no código-fonte da própria aplicação que ocasione um vazamento de memória, drenando esse recurso no servidor.

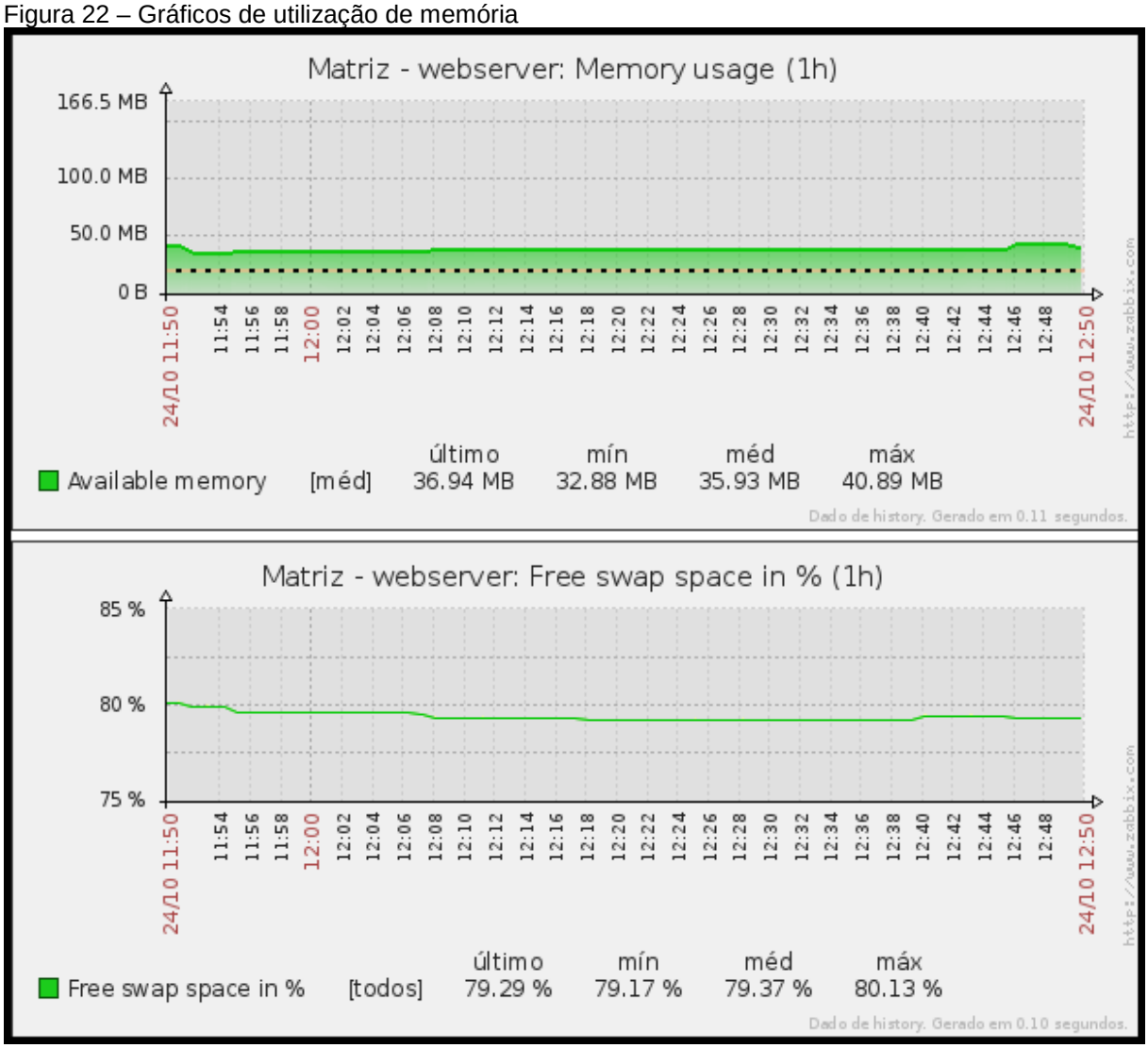
O teste consistiu em executar o software Apache Tomcat[22] em um servidor com pouca memória RAM. O uso do Tomcat deu-se por ele possibilitar ser configurado para que utilize um valor mínimo de memória, causando o consumo quase total desse recurso no servidor, assim que foi iniciado. O resultado pode ser visto na figura 22. No gráfico de uso de memória é possível ver que a quantidade de memória disponível está baixa.

[22] Servidor de aplicação para linguagem de programação Java (SAWAYA, 1999).

Com a diminuição da memória livre disponível, é possível observar que as operações de leitura e escrita no disco perderam desempenho. Analisando o gráfico de memória *Swap*[23] livre, constata-se que o espaço de trocas também está sendo bem utilizado.

O administrador observando os dois gráficos pode pressupor que há correlação entre o baixo desempenho do disco e o autoconsumo das memórias. O gargalo pode estar sendo criado pelo processo de *Swapping*[24] que a gerência de memória do sistema operacional está realizando. Neste caso o Zabbix ajuda o administrador a estabelecer um ponto de partida para corrigir o problema.

Figura 22 – Gráficos de utilização de memória



Fonte: Do autor.

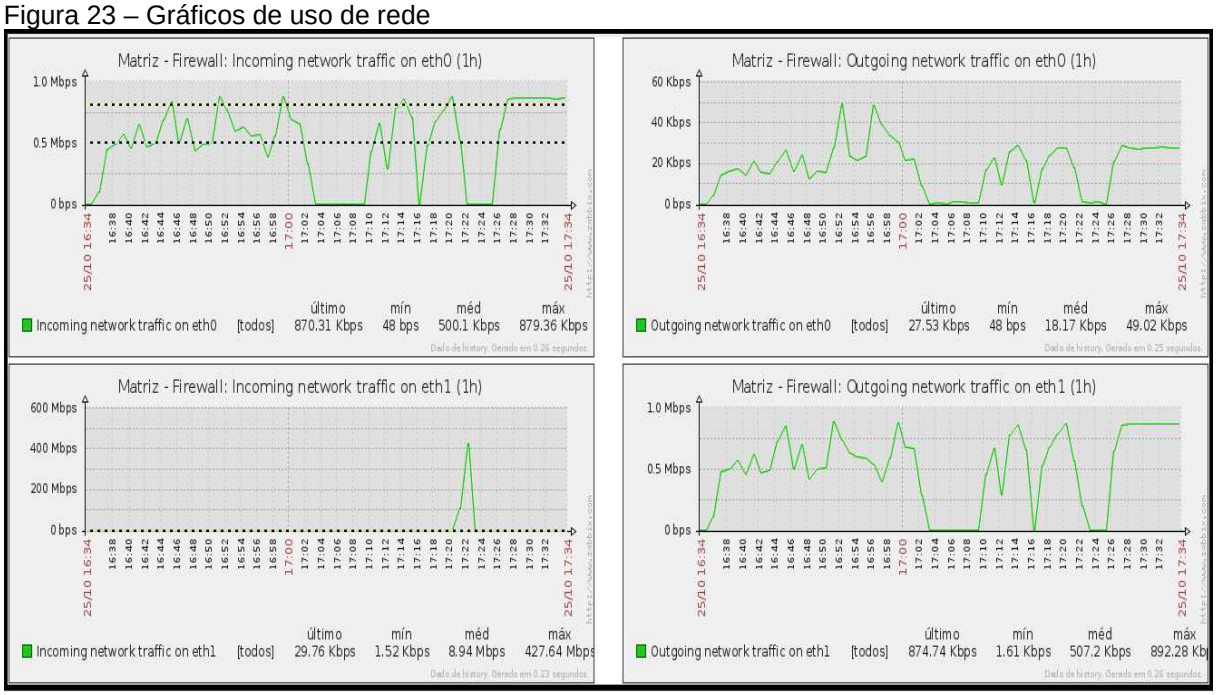
[23]Memória virtual utilizada como armazenamento secundário em disco (Tanenbaum, 2003).
[24]Processo de troca entre memória principal e a memória swap, realizado pelo gerenciador de memória do sistema operacional (Tanenbaum; Woodhull, 2008).

Outra questão problemática recorrente nas redes das empresas é o uso indiscriminado do link de Internet. Ações importantes na empresa podem deixar de ser feitas, por que alguém está fazendo uso errado desse recurso.

O teste feito consistiu em fazer o download de um arquivo de 600MB em um link de Internet de 1Mbps, a escolha do valor deu-se para que a saturação fosse facilmente notada. O limite foi aplicado com o uso de um *proxy HyperText Transfer Protocol* (HTTP[25]). Durante a realização do download foi possível ver nos gráficos indicativos do tráfego de rede, o aumento da quantidade de dados transferidos, visto na figura 23.

É possível observar também que os dados entrando pela interface *Wide Area Network* (WAN) (*Incoming network traffic on eth0*), e saindo pela *Local Area Network* (LAN) (*Outgoing network traffic on eth1*), são semelhantes e indicam que o fluxo de informações atravessa o *firewall*, além de demonstrar o consumo quase total da banda disponível.

Figura 23 – Gráficos de uso de rede



Fonte: Do autor.

Por garantia foi configurado o envio de alertas quando o nível de uso do link chegasse a 50% e 80% do total, para manter o administrador informado mesmo

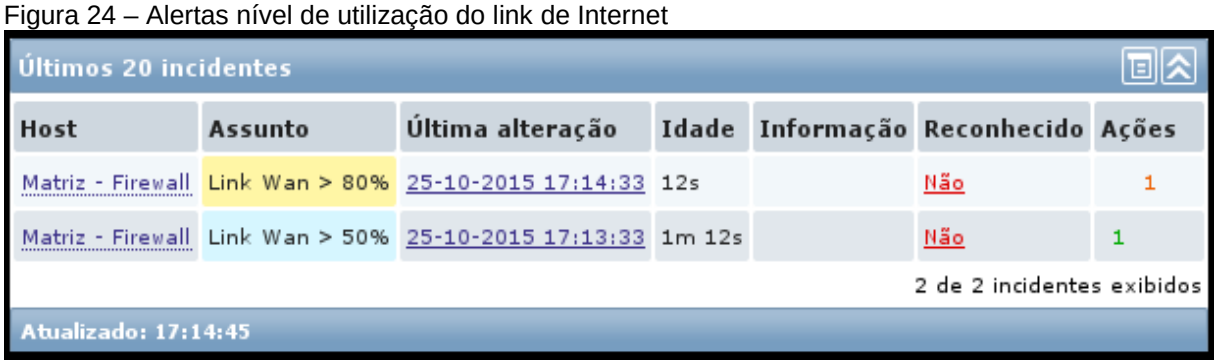
[25]Protocolo utilizado nas transferências de informações formadas pela linguagem HTML (COSTA, 2008).

estando distante da interface de gerenciamento da ferramenta. O resultado dos alertas pode ser observado na figura 24.

As notificações serviram para diminuir ainda mais o tempo de resposta do administrador na análise do incidente.

Sistemas de gestão estão presentes em quase todas as empresas e o armazenamento das informações corporativas geradas é feito em sistemas de gerenciamento de banco de dados.

Figura 24 – Alertas nível de utilização do link de Internet



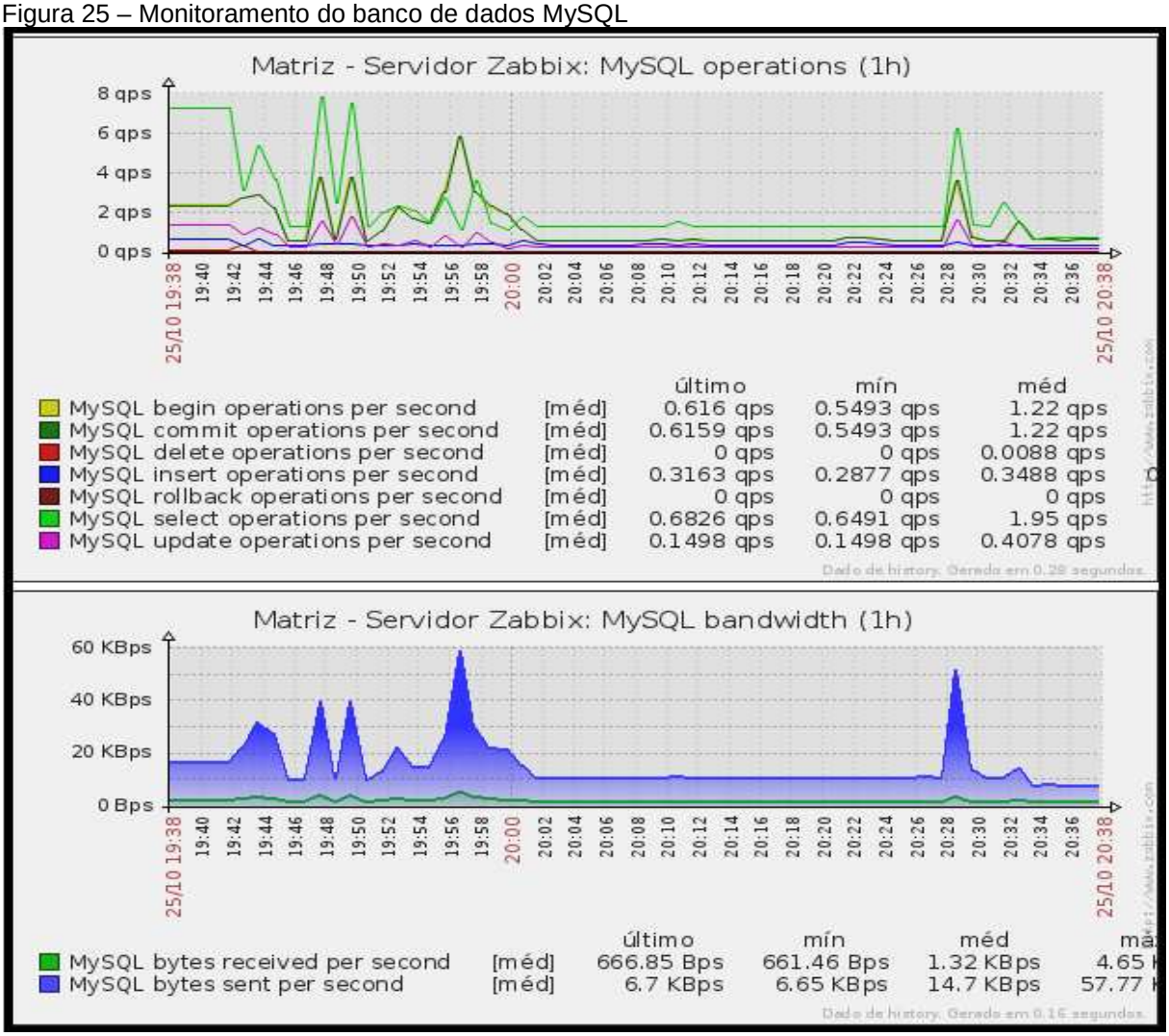
Fonte: Do autor.

O Zabbix em sua instalação padrão tem a capacidade de monitorar o banco de dados MySQL, sendo possível a geração de estatísticas sobre as operações básicas de excluir, inserir, atualizar e buscar dados, além de poder observar a quantidade de dados recebidos e enviados nessas operações. Na figura 25 são demonstrados dois gráficos com essas informações.

Um ponto negativo no monitoramento desse banco de dados é que a ferramenta, na configuração padrão, não possui alertas relacionados com os níveis de utilização. O único alerta disponível é a verificação do serviço, para saber se ele está respondendo.

Para melhorar a aplicação da ferramenta existe na Internet customizações desenvolvidas por usuários, que podem ser integradas ao Zabbix, para melhorar a experiência com o banco de dados. No site oficial, há documentação para que se criem as próprias notificações de forma flexível em seu ambiente, baseadas na coleta de informações feitas por plugins que o administrador poderá criar em qualquer linguagem, de acordo com sua necessidade.

Figura 25 – Monitoramento do banco de dados MySQL



Fonte: Do autor.

Algumas notificações feitas pela ferramenta, relacionadas à segurança nos servidores, puderam ser observadas durante os testes.

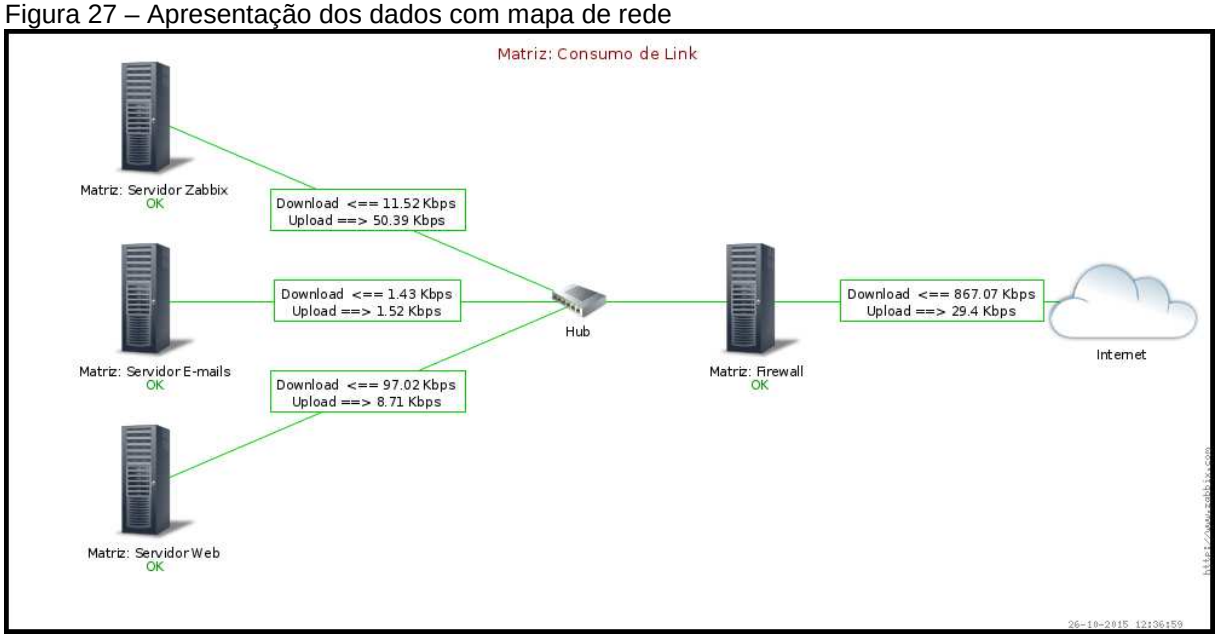
A principal é a verificação de integridade em arquivos. Algumas configurações precisaram ser feitas nos servidores, incluindo a adição de novos usuários. Assim que o procedimento foi realizado, os alertas começaram a aparecer. Pode-se ver um exemplo desses alertas na figura 26, o arquivo */etc/passwd* foi modificado e o Zabbix notificou devidamente o administrador.

Figura 26 – Notificação de perda da integridade de arquivo

Fonte: Do autor.

A apresentação dos dados demonstrou ser um diferencial na ferramenta. Ideal para a criação de centros de operação para controle da infraestrutura de rede em empresas. A customização das informações com a criação de telas personalizadas, a junção das telas em apresentações exibidas em intervalos de tempos e a criação de mapas personalizados de rede tornam esse recurso útil para serem exibidos em telões. Um exemplo de mapa de rede exibindo a infraestrutura da matriz e o consumo dos links de rede pode ser visto a seguir na figura 27.

Figura 27 – Apresentação dos dados com mapa de rede



Fonte: Do autor.

O Zabbix demonstrou possuir inúmeras funcionalidades oferecidas para o monitoramento, alertas e relatórios. De itens de desempenho como opções de nível de utilização e consumo de recursos como CPU, memória, disco, rede, operações em banco de dados além de opções de notificação diversificadas, para garantir que o administrador seja avisado em caso de falhas.

O módulo utilizado para as tarefas administrativas foi projetado seguindo os conceitos das linguagens orientadas a objetos. Dessa forma muitas configurações feitas foram reaproveitadas por meio de herança de propriedades, poupando tempo na configuração de novos itens que vão de perfis semelhantes de servidores monitorados até acessos especiais para usuários diferenciados, com auditoria completa para comprovar qualquer dúvida a respeito de ações realizadas.

Há um controle organizado dos dados armazenados, podendo-se automatizar o processo de limpeza de informações que não são mais relevantes.

A integração com LDAP permite o controle centralizado da autenticação de usuários, ideal no ambiente empresarial.

A apresentação das informações conta com relatórios, incluindo o de disponibilidade, utilizado na avaliação de qualidade dos serviços prestados. É possível ainda fazer o controle de inventário dos servidores gerenciados, útil em empresas que aplicam governança em TI.

A organização dos menus é outro ponto positivo, pois torna a busca das informações simples. Os módulos não apresentaram nenhum problema na instalação e no uso, mostrando que o sistema é estável.

Um ponto negativo é a comunicação não criptografada entre agentes e *proxies* e o servidor Zabbix, podendo haver quebra de segurança nas comunicações pela Internet, quando se monitora redes distribuídas. Isso foi contornado utilizando uma VPN.

Os testes aplicados procuraram demonstrar eventos reais em um ambiente simulado, e os benefícios obtidos pelo administrador no uso da ferramenta. Na instalação padrão os requisitos solicitados foram atendidos podendo ainda ser ampliados de acordo com a necessidade. A exibição dos dados coletados mostrou-se flexível conforme as exigências do administrador.

6 CONCLUSÃO

O uso de ferramentas de gerenciamento é cada vez mais necessário, já não sendo mais possível somente a intervenção manual no ambiente, frente à complexidade enfrentada pelo administrador. Por meio delas é possível saber como anda a saúde da rede, pelo conhecimento das informações que monitoram.

O Zabbix pela robustez e flexibilidade é uma dessas ferramentas capazes de auxiliar o administrador a gerenciar infraestruturas distribuídas.

O estudo demonstrou sua capacidade na detecção de falhas e erros no monitoramento dos ambientes. Diferenciando-se por agregar testes, como no caso da checagem de conteúdo Web e o monitoramento do serviço HTTP, gerando melhores resultados.

Adaptação dos perfis de monitoramento com a detecção automática dos novos itens instalados nos servidores, diminuindo o trabalho do administrador na configuração do sistema.

Possibilidade de ações proativas programadas, como a execução de comandos remotos para reinicialização dos serviços em caso de falha detectada.

Ele ainda comprovou eficiência na antecipação de ações indesejáveis capazes de comprometer a infraestrutura, por meio dos gráficos de tendências de comportamento, feitos a partir dos dados coletados pelos agentes, que orientaram a ação antecipada do administrador, antes do erro ocorrer efetivamente.

O Zabbix demonstrou sua capacidade no gerenciamento de ambientes distribuídos devido o seu módulo *proxy*, que mantém a consistência dos dados coletados, mesmo quando ocorre queda das comunicações.

A apresentação dos dados é feita por gráficos diversificados, que podem ser integrados produzindo telas customizadas que organizam melhor a informação, mapas de rede que servem para visualizar melhor o estado da infraestrutura.

Os testes aplicados no ambiente simulado permitiram a validação de suas funções, atingindo o objetivo de demonstrar sua eficácia auxiliando o administrador no gerenciamento de redes distribuídas.

O ponto negativo encontrado é a ausência de criptografia na comunicação entre seus módulos, que é uma falha de segurança no monitoramento de redes pela Internet. No estudo esse problema foi superado utilizando uma VPN, configurada com o software *OpenVPN*, entre os *firewalls* da matriz e a filial.

A ferramenta somente não substitui o administrador, mas é peça chave para que ele realize um bom trabalho na gestão dos ambientes.

Mais funções poderiam ter sido apresentadas e avaliadas neste trabalho, porém os testes focaram-se somente nas áreas funcionais definidas pela ISO relacionadas à gerência de falhas e desempenho dos serviços e servidores.

A gama de itens envolvidos no gerenciamento de redes é ampla e a ferramenta Zabbix também possui um vasto número de funções que ainda podem ser explorados. Partindo desse principio como sugestão para trabalhos futuros, para ampliar os conhecimentos na ferramenta, realizar estudos aplicando o Zabbix nas demais áreas funcionais da gerência de redes, além das aplicadas neste trabalho.

REFERÊNCIAS

ALVES, Maicon Melo. **Sockets Linux.** Rio de Janeiro: Brasport, 2008.

ARNETT, Matthew F. **Desvendando o TCP/IP:** método de instalação, manutenção e implementação de redes TCP/IP. Rio de Janeiro: Campus, 1997.

BADGER, Michael. **Zenoss Core Network and System Monitoring**. Birmingham: Packt Publishing Ltd, 2008.

BLACK, Tomas Lovis. **Comparação de Ferramentas de Gerenciamento de Redes**. Disponível em: <http://www.lume.ufrgs.br/bitstream/handle/10183/15986/000695315.pdf>. Acesso em: 15 de junho de 2015.

CARDOSO, Leandro Koehler. **Implantação da Ferramenta Nagios para Monitoração de Rede e Análise e Tratamento dos Eventos por Meio de Software de Apoio.** Disponível em: <http://tcc.kironunesc.net.br/arquivos/trabalhos/294.pdf>. Acesso em: 15 de junho de 2015.

CASAD, Joe; WILLSEY, Bob. **Aprenda em 24 horas Tcp/Ip**. Rio de Janeiro: Campus, 1999.

COMER, Douglas E. **Interligação de Redes com TCP/IP**. Rio de Janeiro: Elsevier, 2006.

_____. **Redes De Computadores e Internet**. São Paulo: Bookman, 2007.

COSTA, Daniel G. **JAVA em Rede**. Rio de Janeiro: Brasport, 2008.

COSTA, Daniel G. Administração **de rede com scripts:** Bash, Python e VBScript. Rio de Janeiro: Brasport, 2010.

FARIAS, Paulo César Bento. **Curso Essencial de Redes**. São Paulo: Digerati, 2006.

FARREL, Adrian. **A internet e seus Protocolos:** uma análise comparativa. São Paulo: Campus, 2005.

FOROUZAN, Behrouz A. **Comunicação de dados e redes de computadores**. Porto Alegre: Bookman, 2006.

FOROUZAN, Behrouz A.; FEGAN, Sophia Chung. **Protocolo TCP/IP**. 3. ed. Porto Alegre: AMGH, 2010.

FOROUZAN, Behrouz A.; MOSHARRAF, Firouz. **Redes de Computadores:** Uma abordagem Top-Down. AMGH: Porto Alegre, 2012.

GURGEL, Paulo Henrique Moreira; BRANCO, Kalinka Regina Lucas Castelo; HACHEY, Ghislain. **OpenNMS Starter**. Birmingham: Packt Publishing, 2013.

HORST, Adail Spínola; PIRES, Aécio dos Santos; DÉO, André Luis Boni. **De A a ZABBIX**. São Paulo: Novatec Editora, 2015.

JARGAS, Aurélio Marinho. **Shell Script Profissional**. São Paulo: Novatec, 2008.

JOSEPHSEN, David. **Building a Monitoring Infrastructure with Nagios**. Boston: Pearson Education, 2007.

KUNDU, Dinangkur e LAVLU, S. M. Ibrahim. **Cacti 0.8 Network Monitoring**. Birmingham: Packt Publishing, 2009.

KUROSE, James F.; ROSS, Keith W. **Redes de Computadores e a Internet:** uma abordagem top-down. São Paulo: Pearson Addison Wesley, 2006.

LIMA, Janssen Dos Reis. **Monitoramento de Redes com ZABBIX**. Rio de Janeiro: Brasport, 2014.

MACEDO, Vladimiro Florival Sousa da Rocha Pinto de. **Monitorização de Sistemas de Informação Críticos**. Disponível em: <http://repositorio-aberto.up.pt/bitstream/10216/63403/1/000149170.pdf>. Acesso em: 15 de junho de 2015.

MICROSOFT. Ping. Versão 6.1.7600.16385. [S.l.]: Microsoft Corporation, 2009.

_____. Tracert. Versão 6.1.7600.16385. [S.l.]: Microsoft Corporation, 2009.

_____. Netstat. Versão 6.1.7600.16385. [S.l.]: Microsoft Corporation, 2009.

NAGIOS. Versão 3.5.1. Disponível em: <http://www.nagios.org>. Acesso em: 15 de junho de 2015.

NUNES, Leonardo Batista. **Gerenciamento de uma Rede de Computadores em um Ambiente Corporativo (UEPB/Campus VIII) Utilizando o Software Zabbix**. Disponível em:

<http://dspace.bc.uepb.edu.br:8080/jspui/bitstream/123456789/3052/1/PDF%20-%20Leonardo%20Batista%20Nunes.pdf>. Acesso em: 15 de junho de 2015.

OLIVEIRA, Mírian; ABDALA, Elisabeth Avila. **Tecnologias da Internet**: Casos Práticos em Empresas. Porto Alegre: Edipucrs, 2003.

OPENNMS. 2015. Versão 16.0.0. Disponível em: <http://www.opennms.org>. Acesso em: 15 de junho de 2015.

PITANGA, Marcos. **Construindo Supercomputadores com Linux**. Rio de Janeiro: Brasport, 2008.

PÉRICAS, Francisco Adell. **Redes de Computadores**: Conceitos e a Arquitetura Internet. Blumenal: Edição do autor, 2012.

PUSKA, Alisson Andrey. **Solução de Gerenciamento de Redes Utilizando o Sistema de Código Aberto:** Zabbix. Disponível em: <http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/395/1/CT_GESER_1_2011_04.pdf>. Acesso em: 15 jun 2015.

ROSS, Júlio. **Redes de Computadores**. São Paulo: Antenna Edições Técnicas, 2008.

SAWAYA, Márcia Regina. **Dicionário de Informática e Internet**. São Paulo: Nobel, 1999.

SCHMITT, Marcelo Augusto Rauh; PERES, André; LOUREIRO, César Augusto Hass. **Redes de Computadores**: Nível de Aplicação e Instalação de Serviços. Porto Alegre: Bookman, 2013.

STALLINGS, Willian. **Redes e sistemas de comunicação de dados**. Rio de Janeiro: Elsevier, 2005.

STATDLOBER, Juliano. **Help-Desk e SAC com Qualidade**. Rio de Janeiro: Brasport, 2006.

STATO FILHO, André. **Linux controle de redes**. Florianópolis: Visual Books, 2009.

STEVENS, W. Richard; FENNER Bill; RUDOFF, Andrew M. **Programação de rede UNIX:** Api para soquetes de rede, Porto Alegre: Bookman, 2005.

STRANGER, James; LANE, Patrick. **Hack Proofing Linux:** A Guide to Open Source Security. Rockland: Syngress, 2001.

TANENBAUM, Andrew S. **Redes de Computadores**. Rio de Janeiro: Elsevier, 2003.

TANENBAUM, Andrew S.; Woodhull, Albert S. Sistemas Operacionais: Projeto e Implementação. Rio de Janeiro: Artmed, 2008.

TCPDUMP. Versão 4.7.4. Disponível em: <http://www.tcpdump.org>. Acesso em: 15 de junho de 2015.

TEIXEIRA JÚNIOR, José Helvécio; SUAVÉ, Jacques Philippe; MOURA, José Antão Beltrão; TEIXEIRA, Suzana de Queiroz Ramos. **Redes de Computadores:** Serviços, Administração e Segurança. São Paulo: Makron-Books, 1999.

TURNBULL, James. **Pro Nagios 2.0**. New York: Apresss, 2006.

VACCHE, Andrea Dalle; LEE, Stefano Kewan. **Zabbix Network Monitoring Essentials**. Birmingham: Packt Publishing, 2015.

VELLOSO, Fernando de Castro. **Informática Básica Conceitos**. Rio de Janeiro: Elsevier, 2014.

WEILL, Peter; ROSS, Jeanne W. **Conhecimento em TI**. São Paulo: M. Books,2010

WIRESHARK. Versão 1.12.5. Disponível em: <http://www.wireshark.org>. Acesso em: 15 de junho de 2015.

ZABBIX . Versão 2.2 LTS. Disponível em: <http://www.zabbix.org>. Acesso em: 15 de junho de 2015.

ZENOSS. Versão 5 beta 3. Disponível em: <http://www.zenoss.org>. Acesso em: 15 de junho de 2015.

APÊNDICE(S)

APÊNDICE A – AMBIENTE

1 INSTALAÇÃO DO HOSPEDEIRO - UBUNTU 14.04

O processo de instalação do Ubuntu segue os seguintes passos:

a) rode seu *live-DVD* ou pendrive, clique em *Try Ubuntu* e, na área de trabalho, clique em *Install Ubuntu-14.04-LTS*;
b) selecione o idioma: a opção *English* foi a escolhida por ser a padrão;
c) certifique-se se ter espaço suficiente em disco, de estar conectado numa fonte de energia e na Internet e clique em *Continue*;
d) selecione *Erase disk and install Ubuntu* e clique em *Install Now*;
e) selecione *Write the changes to disks*, confirme clicando em *Continue*;
f) selecione o fuso horário: *Sao Paulo* e clique em *Continue*;
g) selecione o leiaute do teclado: *Portuguese Brazil* e clique em *Continue*;
h) preencha os dados para a criação de um usuário sem privilégios, incluindo a senha e clique em *Continue*;
i) aguarde o processo de instalação concluir;
j) reinicie seu computador.

1.1 CONFIGURAÇÃO DO UBUNTU

A configuração do Ubuntu começa pelo gerenciamento de pacotes. Para gerencia-los ele utiliza a ferramenta APT, que centraliza as informações sobre repositórios no arquivo *sources.list*. Como primeiro passo foram eliminados os repositórios desnecessários, sobraram somente os repositórios para pacotes binários em versões estáveis. O editor de textos *vi* é padrão nos terminais GNU/Linux e foi utilizado na tarefa. É necessário abrir um terminal de comandos como superusuário *root* para digitar os comandos, sempre que for solicitado que se digite um comando será utilizado esse mesmo terminal para isso, o resultado é visto a seguir:

vi /etc/apt/sources.list

deb http://55.archive.ubuntu.com/ubuntu/ trusty main restricted universe multiverse

deb http://55.archive.ubuntu.com/ubuntu/ trusty-security main restricted universe multiverse
deb http://55.archive.ubuntu.com/ubuntu/ trusty-updates main restricted universe multiverse
deb http://55.archive.ubuntu.com/ubuntu/ trusty-proposed main restricted universe multiverse
deb http://55.archive.ubuntu.com/ubuntu/ trusty-backports main restricted universe multiverse

Após é necessária a atualização da lista de pacotes e a atualização dos pacotes já instalados no sistema para a última versão disponível. O software *aptitude* foi utilizado ao invés do *apt-get*, padrão do sistema APT por resolver melhor questões de dependências na instalação e remoção de pacotes. Digite os comandos a seguir:

aptitude update
aptitude –y safe-upgrade

2 INSTALAÇÃO DO ORACLE VM VIRTUALBOX

Para a instalação do software virtualizador e o restante do ambiente será necessário um link de Internet, primeiro é preciso baixá-lo antes, a versão estável disponível durante a realização da pesquisa é a 4.3.30. Para fazer o download e a instalação digite os comandos:

wget http://download.virtualbox.org/virtualbox/4.3.30/virtualbox-4.3_4.3.30-101610~Ubuntu~quantal_amd64.deb

dpkg -i virtualbox-4.3_4.3.30-101610~Ubuntu~quantal_amd64.deb

2.1 CRIAÇÃO DAS MÁQUINAS VIRTUAIS

Para a criação da máquina virtual base, que será utilizada em todos os servidores do ambiente, siga os seguintes passos:

a) na tela principal do Oracle VirtualBox clique em *New;*
b) em *Name and operating system*, coloque um nome, em *type* escolha: *Linux*, em *version*: *Debian (64-bit)* e clique em *Next*;
c) em *Memory size*, preencha com o valor 512 MB e clique em *Next*;
d) em *Hard drive*, escolha *Create a virtual hard drive now* e clique em *Create*;

e) em *Hard drive file type*, escolha: *VDI (VirtualBox Disk Image)* e clique em *Next*;
f) em *Storage on physical hard drive*, escolha: *Dinamically allocated* e clique em *Next*;
g) em *File location and size*, preencha com o valor: 8.00GB e clique em *Create*.

3 INSTALAÇÃO DO DEBIAN

É necessário realizar o download do arquivo antes da instalação. No terminal no Ubuntu digite:

wget http://cdimage.debian.org/debian-cd/8.2.0/amd64/iso-cd/debian-8.2.0-amd64-netinst.iso

A instalação do Debian que servirá como base para as demais máquinas, pode ser realizada somente com a imagem ISO sem necessidade de gravá-la em CD ou *pen-drive*. Inicie a máquina virtual criada anteriormente, e selecione o arquivo *debian-8.2.0-amd64-netinst.iso* na janela *Select a start-up disk*, clique em *OK* para a instalação iniciar e siga os seguintes passos:

a) selecione *Install* e tecle *ENTER*;
b) em *Select a language* escolha *English* e tecle *ENTER*;
c) em *Select your location* escolha *other*, *South America* e *Brazil* e tecle *ENTER*;
d) em *Configure locales* escolha *United States* e tecle *ENTER*;
e) em *Configure the keyboard* escolha *Brazilian* e tecle *ENTER*;
f) em *Configure the network* digite o *hostname* e tecle *ENTER*, em *Domain name* tecle *ENTER*;
g) em *Set up users and password* digite a senha para o usuário *root* e tecle *ENTER*. Repita a senha e tecle *ENTER*;
h) em *Configure the clock* selecione *São Paulo* e tecle *ENTER*;
i) em *Partition disks* selecione *Guided – use entire disk* e tecle *ENTER*, a seguir escolha *All files in one partition* e tecle *ENTER*, depois escolha

Finish partitioning and write changes to disk e tecle *ENTER*, por último selecione a opção *Yes* em *Write the changes to disk*;

j) em *Configure the package manager* selecione *Brazil* e tecle *ENTER*, selecione *ftp.br.debian.org* e tecle *ENTER*;

k) em *Software selection* deixar somente a opção *SSH Server* e tecle *ENTER*;

l) aguarde o processo de instalação prosseguir;

m) em *Install the GRUB boot loader on a hard disk* escolha *Yes* e tecle *ENTER*;

n) reinicie o servidor.

3.1 CONFIGURAÇÃO DEBIAN

A configuração do Debian também começa pelo gerenciamento de pacotes. É necessário fazer o *login* como usuário *root* e editar o arquivo */etc/apt/sources.list*, deixando somente os repositórios principais, igual o exemplo:

```
vi /etc/apt/sources.list
------------------------------------------------------------
deb http://ftp.br.debian.org/debian/ jessie main
deb http://security.debian.org/ jessie/updates main
deb http://ftp.br.debian.org/debian/ jessie-updates main
------------------------------------------------------------
```

Antes de atualizar o ambiente é necessário instalar alguns softwares no Debian. A instalação feita foi para um perfil mínimo será necessário utilizar a ferramenta padrão *apt-get*, digitando o comando:

```
apt-get install -y aptitude vim less rcconf stress
```

Para a atualização da lista de pacotes e do sistema operacional digite os dois comandos:

```
aptitude update
aptitude –y safe-upgrade
```

Para facilitar a configuração dos módulos do Zabbix, habilitaremos a sintaxe colorida no editor:

```
vim /etc/vim/vimrc
-------------------------
syntax on
-------------------------
```

Como esta máquina será base para as demais da infraestrutura, o repositório do Zabbix será instalado e configurado agora. Faça o download do arquivo com as informações do repositório, instale-o e atualize a lista de pacotes:

```
wget http://repo.zabbix.com/zabbix/2.4/debian/pool/main/z/zabbix-release/zabbix-release_2.4-1+jessie_all.deb

dpkg -i zabbix-release_2.4-1+jessie_all.deb

aptitude update
```

Para continuar será necessário desligar o Debian base, com o comando:

```
init 0
```

4 CRIAÇÃO DO AMBIENTE

A seguir serão criadas e configuradas as máquinas para que formem os ambientes simulados da matriz e da filial.

4.1 Matriz – SERVIDOR ZABBIX

O primeiro passo é clonar a máquina base no *VirtualBox*, como nome escolha Matriz - ZabbixServer e marque *Reinitialize the MAC address of all network cards*, clique em *Next*, selecione *Full clone* e clique no botão *Clone*. Edite as configurações da máquina, e em *Network*, *Adapter 1* altere a opção *Attached to:* para *Internal network*, clique em *Ok* para concluir e inicie a máquina.

Efetue *login* como *root* e altere *hostname* e as configurações de rede seguindo os exemplos:

vim /etc/hostname
---
zabbixservermatriz
---

vim /etc/hosts

127.0.0.1 localhost
127.0.1.1 zabbixservermatriz
 # The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
---

vim /etc/network/interfaces
---
*source /etc/network/interfaces.d/**
auto lo
iface lo inet loopback
allow-hotplug eth0
iface eth0 inet static
address 10.10.10.10
netmask 255.255.255.0
gateway 10.10.10.1
---

vim /etc/resolv.conf
---
nameserver 8.8.8.8
nameserver 8.8.4.4

4.2 MATRIZ – SERVIDOR DE BANCO DE DADOS

Clonar a máquina base no *VirtualBox*, como nome escolha *Matriz - DBServer* e marque *Reinitialize the MAC address of all network cards*, clique em *Next*, selecione *Full clone* e clique no botão *Clone*. Edite as configurações da máquina, e em *Network*, *Adapter 1* altere a opção *Attached to:* para *Internal network*, clique em *Ok* para concluir e inicie a máquina.

Efetue *login* como *root* e altere *hostname* e as configurações de rede seguindo os exemplos:

vim /etc/hostname
---
dbservermatriz
---

vim /etc/hosts

```
127.0.0.1       localhost
127.0.1.1       dbservermatriz
# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
--------------------

vim /etc/network/interfaces
--------------------
source /etc/network/interfaces.d/*
auto lo
iface lo inet loopback
allow-hotplug eth0
iface eth0 inet static
        address 10.10.10.11
        netmask 255.255.255.0
        gateway 10.10.10.1
--------------------

vim /etc/resolv.conf
-------------------------
nameserver 8.8.8.8
nameserver 8.8.4.4
-------------------------
```

4.3 MATRIZ - SERVIDOR HTTP

Clonar a máquina base no *VirtualBox*, como nome escolha *Matriz - WebServer* e marque *Reinitialize the MAC address of all network cards*, clique em *Next*, selecione *Full clone* e clique no botão *Clone*. Edite as configurações da máquina, e em *Network*, *Adapter 1* altere a opção *Attached to:* para *Internal network*, clique em *Ok* para concluir e inicie a máquina.

Efetue *login* como *root* e altere *hostname* e as configurações de rede seguindo os exemplos:

```
vim /etc/hostname
----------------------
webservermatriz
--------------------

vim /etc/hosts
--------------------
127.0.0.1       localhost
127.0.1.1       webservermatriz
# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
--------------------

vim /etc/network/interfaces
```

```
----------------------------------
source /etc/network/interfaces.d/*
auto lo
iface lo inet loopback
allow-hotplug eth0
iface eth0 inet static
        address 10.10.10.12
        netmask 255.255.255.0
        gateway 10.10.10.1
----------------------------------

vim /etc/resolv.conf
-------------------------
nameserver 8.8.8.8
nameserver 8.8.4.4
-------------------------
```

4.4 MATRIZ – SERVIDOR DE E-MAIL

Clonar a máquina base no *VirtualBox*, como nome escolha *Matriz - MxServer* e marque *Reinitialize the MAC address of all network cards*, clique em *Next*, selecione *Full clone* e clique no botão *Clone*. Edite as configurações da máquina, e em *Network*, *Adapter 1*, altere a opção *Attached to:* para *Internal nework* selecione clique em *Ok* para concluir e inicie a máquina.

Efetue *login* como *root* e altere *hostname* e as configurações de rede seguindo os exemplos:

```
vim /etc/hostname
--------------------
mxservermatriz
--------------------

vim /etc/hosts
--------------------
127.0.0.1       localhost
127.0.1.1       mxservermatriz

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
--------------------

vim /etc/network/interfaces
----------------------------------
source /etc/network/interfaces.d/*
auto lo
iface lo inet loopback
allow-hotplug eth0
iface eth0 inet static
        address 10.10.10.13
```

```
        netmask 255.255.255.0
        gateway 10.10.10.1
-----------------------------------

vim /etc/resolv.conf
------------------------
nameserver 8.8.8.8
nameserver 8.8.4.4
------------------------
```

4.5 MATRIZ – FIREWALL

Clonar a máquina base no *VirtualBox*, como nome escolha *Matriz - Firewall* e marque *Reinitialize the MAC address of all network cards*, clique em *Next*, selecione *Full clone* e clique no botão *Clone*. Edite as configurações da máquina, e em *Network*, *Adapter 1*, altere a opção *Attached to:* para *Bridged Adapter,* essa interface será a WAN. Em *Adapter 2*, Marque a opção *Enable Network Adapter*, altere a opção *Attached to:* para *Internal Nework,* essa será a LAN, selecione clique em *Ok* para concluir e inicie a máquina.

Efetue *login* como *root* e altere *hostname* e as configurações de rede seguindo os exemplos:

```
vim /etc/hostname
----------------------
firewallmatriz
----------------------

vim /etc/hosts
--------------------
127.0.0.1       localhost
127.0.1.1       firewallmatriz
# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
--------------------

vim /etc/network/interfaces
-----------------------------------
source /etc/network/interfaces.d/*
auto lo
iface lo inet loopback
allow-hotplug eth0
allow-hotplug eth0
iface eth0 inet static
        address 192.168.1.100
        netmask 255.255.255.0
        gateway 192.168.1.1

allow-hotplug eth1
```

```
iface eth1 inet static
        address 10.10.10.1
        netmask 255.255.255.0
-----------------------------------

vim /etc/resolv.conf
------------------------
nameserver 8.8.8.8
nameserver 8.8.4.4
------------------------
```

Para criar o ambiente de uma rede protegida por *firewall*, é necessário configurar o roteamento. Um novo pacote será instalado: "*iptables-persistent*", para armazenar as configurações de roteamento no *iptables* (interface de configuração do filtro de pacotes do GNU/Linux), digitando os comandos:

```
vim /etc/sysctl.conf
----------------------------
net.ipv4.ip_forward=1
----------------------------

sysctl -p

iptables -t nat  -A POSTROUTING -s 10.10.10.0/24 -j -to 192.168.1.100

aptitude install iptables-persistent
```

4.6 FILIAL – ZABBIX PROXY

Clonar a máquina base no *VirtualBox*, como nome escolha *Matriz - ZabbixProxy* e marque *Reinitialize the MAC address of all network cards*, clique em *Next*, selecione *Full clone* e clique no botão *Clone*. Edite as configurações da máquina, e em *Network*, *Adapter 1* altere a opção *Attached to:* para *Internal network*, clique em *Ok* para concluir e inicie a máquina.

Efetue *login* como *root* e altere *hostname* e as configurações de rede seguindo os exemplos:

```
vim /etc/hostname
----------------------
zabbixproxyfilial
-------------------

vim /etc/hosts
--------------------
127.0.0.1       localhost
127.0.1.1       zabbixproxyfilial
# The following lines are desirable for IPv6 capable hosts
```

```
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
--------------------

vim /etc/network/interfaces
--------------------
source /etc/network/interfaces.d/*
auto lo
iface lo inet loopback
allow-hotplug eth0
iface eth0 inet static
        address 20.20.20.10
        netmask 255.255.255.0
        gateway 20.20.20.1
--------------------

vim /etc/resolv.conf
------------------------
nameserver 8.8.8.8
nameserver 8.8.4.4
------------------------
```

4.7 FILIAL – SERVIDOR DE BANCO DE DADOS

Clonar a máquina base no *VirtualBox*, como nome escolha *Filial - DBServer* e marque *Reinitialize the MAC address of all network cards*, clique em *Next*, selecione *Full clone* e clique no botão *Clone*. Edite as configurações da máquina, e em *Network*, *Adapter 1* altere a opção *Attached to:* para *Internal network*, clique em *Ok* para concluir e inicie a máquina.

Efetue *login* como *root* e altere *hostname* e as configurações de rede seguindo os exemplos:

```
vim /etc/hostname
--------------------
dbserverfilial
--------------------

vim /etc/hosts
--------------------
127.0.0.1       localhost
127.0.1.1       dbserverfilial
# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
--------------------

vim /etc/network/interfaces
--------------------
source /etc/network/interfaces.d/*
```

```
auto lo
iface lo inet loopback
allow-hotplug eth0
iface eth0 inet static
        address 20.20.20.11
        netmask 255.255.255.0
        gateway 20.20.20.1
--------------------

vim /etc/resolv.conf
-------------------------
nameserver 8.8.8.8
nameserver 8.8.4.4
-------------------------
```

4.8 FILIAL – SERVIDOR HTTP

Clonar a máquina base no *VirtualBox*, como nome escolha *Filial - WebServer* e marque *Reinitialize the MAC address of all network cards*, clique em *Next*, selecione *Full clone* e clique no botão *Clone*. Edite as configurações da máquina, e em *Network*, *Adapter 1* altere a opção *Attached to:* para *Internal network*, clique em *Ok* para concluir e inicie a máquina.

Efetue *login* como *root* e altere *hostname* e as configurações de rede seguindo os exemplos:

```
vim /etc/hostname
-----------------------
webserverfilial
--------------------

vim /etc/hosts
--------------------
127.0.0.1       localhost
127.0.1.1       webserverfilial
# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
--------------------

vim /etc/network/interfaces
-----------------------------------
source /etc/network/interfaces.d/*
auto lo
iface lo inet loopback
allow-hotplug eth0
iface eth0 inet static
        address 20.20.20.12
        netmask 255.255.255.0
        gateway 20.20.20.1
-----------------------------------
```

vim /etc/resolv.conf
---
nameserver 8.8.8.8
nameserver 8.8.4.4

4.9 FILIAL – SERVIDOR FTP

Clonar a máquina base no *VirtualBox*, como nome escolha *Matriz - FtpServer* e marque *Reinitialize the MAC address of all network cards*, clique em *Next*, selecione *Full clone* e clique no botão *Clone*. Edite as configurações da máquina, e em *Network*, *Adapter 1*, altere a opção *Attached to:* para *Internal Nework* selecione clique em *Ok* para concluir e inicie a máquina.

Efetue *login* como *root* e altere *hostname* e as configurações de rede seguindo os exemplos:

vim /etc/hostname
---
ftpserverfilial
---

vim /etc/hosts

127.0.0.1 localhost
127.0.1.1 ftpserverfilial

 # The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
---

vim /etc/network/interfaces
---
*source /etc/network/interfaces.d/**
auto lo
iface lo inet loopback
allow-hotplug eth0
iface eth0 inet static
address 20.20.20.13
netmask 255.255.255.0
gateway 20.20.20.1
---

vim /etc/resolv.conf
---
nameserver 8.8.8.8
nameserver 8.8.4.4

4.10 FILIAL – FIREWALL

Clonar a máquina base no *VirtualBox*, como nome escolha *Matriz - Firewall* e marque *Reinitialize the MAC address of all network cards*, clique em *Next*, selecione *Full clone* e clique no botão *Clone*. Edite as configurações da máquina, e em *Network*, *Adapter 1*, altere a opção *Attached to:* para *Bridged Adapter,* essa será a interface WAN. Em *Adapter 2*, Marque a opção Enable Network Adapter, altere a opção *Attached to:* para *Internal nework,* essa a interface LAN*,* selecione clique em *Ok* para concluir e inicie a máquina.

Efetue *login* como *root* e altere *hostname* e as configurações de rede seguindo os exemplos:

vim /etc/hostname
----
firewallfilial
----

vim /etc/hosts

127.0.0.1 localhost
127.0.1.1 firewallfilial
 # The following lines are desirable for IPv6 capable hosts
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
----

vim /etc/network/interfaces
----
*source /etc/network/interfaces.d/**
auto lo
iface lo inet loopback
allow-hotplug eth0
iface eth0 inet static
address 192.168.1.200
netmask 255.255.255.0
gateway 192.168.1.1
allow-hotplug eth1
iface eth1 inet static
address 20.20.20.1
netmask 255.255.255.0
----

vim /etc/resolv.conf
----
nameserver 8.8.8.8
nameserver 8.8.4.4

Para criar o ambiente de uma rede protegida por firewall na filial, é necessário configurar o roteamento. Um novo pacote será instalado: "*iptables-persistent"*, para armazenar as configurações de roteamento no *iptables* (interface de configuração do filtro de pacotes do GNU/Linux), digitando os comandos:

vim /etc/sysctl.conf
----
net.ipv4.ip_forward=1
----

sysctl -p

iptables -t nat -A POSTROUTING -s 20.20.20.0/24 -j --to 192.168.1.200

aptitude install iptables-persistent

4.11 VPN

A interligação das redes da Matriz e da Filial será feita via VPN utilizando o software *Openvpn*. Os comandos necessários para instalação e configuração seguem a seguir:

4.11.1 Instalação do servidor (matriz - firewall)

aptitude install openvpn openssl easy-rsa

[Criação das chaves de criptografia]
mkdir /etc/openvpn/easy-rsa/
cd /etc/openvpn/easy-rsa/
cp -R /usr/share/easy-rsa/ .*
cd /etc/openvpn/easy-rsa/

[Alterar locais]
vim vars
---
export KEY_COUNTRY="BR"
export KEY_PROVINCE="SC"
export KEY_CITY="Criciuma"
export KEY_ORG="UNESC"
export KEY_EMAIL="sivasmartins@gmail.com
export KEY_OU="TCC3"

. ./vars
./clean-all
./build-ca
./build-key-server server
./build-dh

#Chaves clientes
./build-key filial

cd /etc/openvpn
cp easy-rsa/keys/ca.crt .
cp easy-rsa/keys/server.key .
cp easy-rsa/keys/server.crt .
cp easy-rsa/keys/dh2048.pem .

vim /etc/openvpn/server.conf
----
ca /etc/openvpn/ca.crt
cert /etc/openvpn/server.crt
key /etc/openvpn/server.key

dh /etc/openvpn/dh2048.pem
dev tun
server 10.8.0.0 255.255.255.0
proto udp
port 1194
#DNS Server
push "dhcp-option DNS 8.8.8.8"
push "dhcp-option DNS 8.8.4.4"
ifconfig-pool-persist ipp.txt
user nobody
group nogroup
persist-key
persist-tun
status /var/log/openvpn/matriz-status.log
log-append /var/log/openvpn/matriz.log
comp-lzo
client-to-client

#Rotas
up /etc/openvpn/rotas.up
down /etc/openvpn/rotas.down
----

vim ipp.txt

#Reserva de endereços
filial,10.8.0.10
----

vim /etc/openvpn/rotas.up

#!/bin/bash
route add -net 20.20.20.0 netmask 255.255.255.0 gw 10.8.0.1 dev tun0
----

chmod 777 /etc/openvpn/rotas.down
vim /etc/openvpn/rotas.down

#!/bin/bash
route del -net 20.20.20.0 netmask 255.255.255.0 gw 10.8.0.1 dev tun0
----

chmod 777 /etc/openvpn/rotas.down
vim /etc/sysctl.conf
----
net.ipv4.ip_forward = 1

*---------- *

sysctl -p

iptables -t nat -A POSTROUTING -s 10.10.10.0/24 -j SNAT --to 10.8.0.1

/etc/init.d/openvpn restart

#Exportar chaves para filial
scp /etc/openvpn/easy-rsa/keys/ca.crt 192.168.1.200:/etc/openvpn
scp /etc/openvpn/easy-rsa/keys/filial.crt 192.168.1.200:/etc/openvpn
scp /etc/openvpn/easy-rsa/keys/filial.key 192.168.1.200:/etc/openvpn

4.11.2 Instalação do cliente (filial - firewall)

aptitude install openvpn

vim /etc/openvpn/client.conf
----
client
dev tun
port 1194
proto udp
remote 192.168.1.100 1194
nobind
ca /etc/openvpn/ca.crt
cert /etc/openvpn/filial.crt
key /etc/openvpn/filial.key

status /var/log/openvpn/filial-status.log
log-append /var/log/openvpn/filial.log

comp-lzo
persist-key
persist-tun

verb 3

#Rotas
up /etc/openvpn/rotas.up
down /etc/openvpn/rotas.down
----

vim /etc/openvpn/rotas.up

#!/bin/bash
route add -net 10.10.10.0 netmask 255.255.255.0 gw 10.8.0.10 dev tun0
----

chmod 777 /etc/openvpn/rotas.down

vim /etc/openvpn/rotas.down

#!/bin/bash
route del -net 10.10.10.0 netmask 255.255.255.0 gw 10.8.0.10 dev tun0
----

chmod 777 /etc/openvpn/rotas.down

vim /etc/sysctl.conf
---
net.ipv4.ip_forward = 1
---

sysctl -p

iptables -t nat -A POSTROUTING -s 20.20.20.0/24 -j SNAT --to 10.8.0.10

/etc/init.d/openvpn restart

4.12 SERVIÇOS

Segue os comandos para instalação dos serviços que serão monitorados respectivamente em cada servidor da infraestrutura.

4.12.1 Matriz - Servidor de banco de dados

aptitude install -y mysql-server

4.12.2 Matriz - Servidor web

aptitude install -y apache2 tomcat8

4.12.3 Matriz - Servidor de e-mail

aptitude install -y postfix dovecot-pop3d dovecot-imapd

4.12.4 Filial - Servidor de banco de dados

aptitude install -y mysql-server

4.12.5 Filial - Servidor web

aptitude install -y apache2 tomcat8

4.12.6 Filial - Servidor ftp

aptitude install -y vsftpd

APÊNDICE B – MONITORAMENTO

1 ZABBIX

1.1 SERVIDOR ZABBIX

Para comodidade na realização das configurações de monitoramento do Zabbix com o navegador de Internet, será necessário a instalação da interface gráfica *Mate* na máquina *Matriz – Servidor Zabbix*.

Primeiro altere as configurações da máquina virtual no VirtualBox para adicionar mais memória, vá em *Settings*, *System*, *Base Memory* e mude o valor para 1536Gb e clique em *Ok*. Após inicie a máquina, faça *login* como usuário *root* e siga os passos seguintes:

aptitude install -y task-mate-desktop

Quando o processo concluir será necessário reinicia-la. Para instalar o módulo servidor do Zabbix, faça *login* como *root* novamente e digite o comando:

aptitude install -y zabbix-server-mysql zabbix-get zabbix-frontend-php

Durante o processo as dependências serão instaladas automaticamente. Algumas informações são solicitadas:

a) na tela *Configuring mysql-server-5.5*, digite uma senha para o administrador do banco dados e tecle *ENTER*, logo a seguir confirme a mesma senha;
b) em *Configuring zabbix-server-mysql*, escolha *Yes* para que o Zabbix crie o banco de dados automaticamente;
c) a seguir digite uma senha para o usuário Zabbix no banco de dados. Essa senha será usada para a interface gráfica se conectar ao banco de dados. Confirme-a em seguida.

Altere o arquivo *php.ini* e inclua os valores a seguir. Esses valores são necessários para o funcionamento da interface Web, após o serviço do apache precisará ser reiniciado.

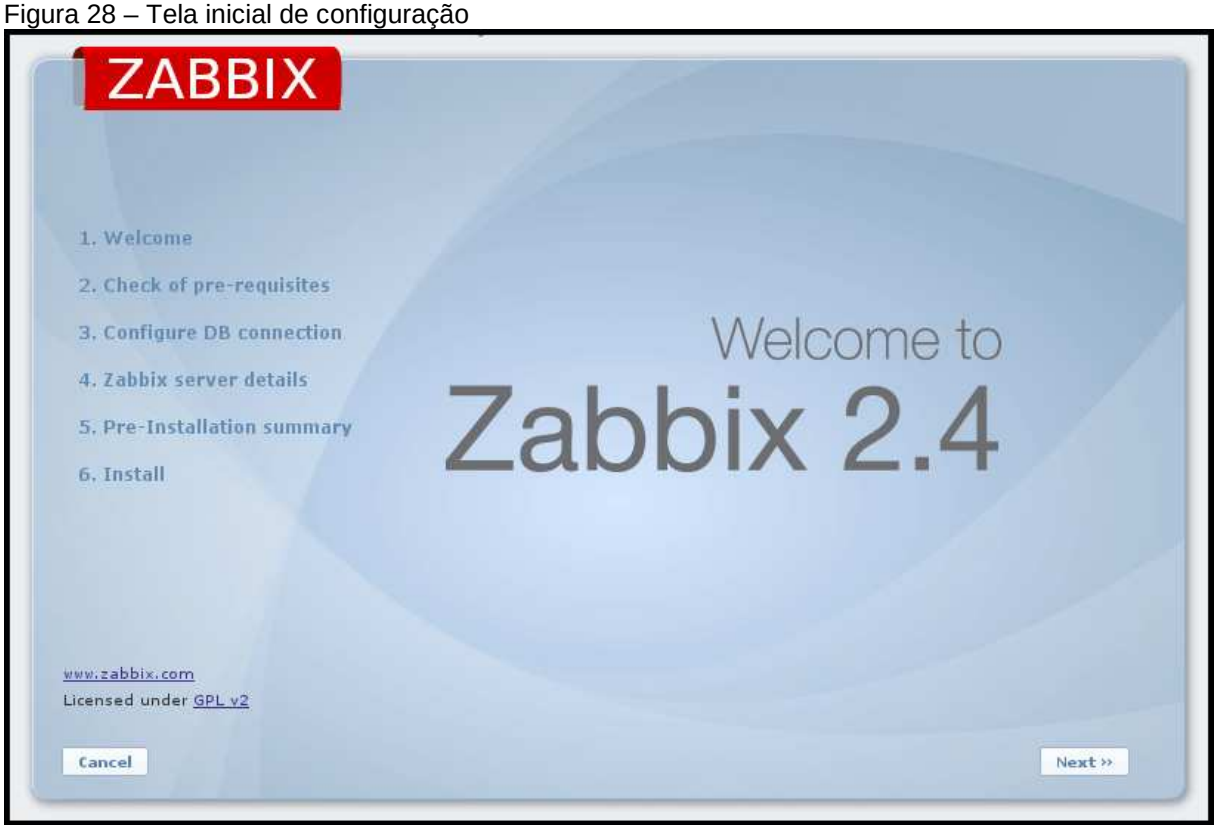
vim /etc/php5/apache2/php.ini
---
date.timezone = America/Sao_Paulo
always_populate_raw_post_data = -1
---

dpkg-reconfigure locales
(escolher as liguagens: pt_BR ISO-8859-1, pt_BR.UTF-8 UTF-8)

/etc/init.d/apache2 restart

A continuação da instalação é via interface Web. Acesse o endereço *"http://10.10.10.10/zabbix"*, o resultado pode ser visto na figura 28, siga os passos para concluir a instalação.

Figura 28 – Tela inicial de configuração



Fonte: Do autor.

a) clique em *Next* na tela inicial;

b) verifique se todos os pré-requisitos estão com um *Ok* e clique em *Next*;

c) preencha as informações para a interface acessar o banco de dados: *Database type: MySQL, Database host: localhost, Database port: 0, Database name: Zabbix, User: zabbix, Password: conforme escolhido*;
d) clique no botão *Test connection* para verificar se o acesso funcionou. Caso afirmatico, continue até finalizar: *Next*, *Next* e *Finish*.

Após concluir a tela de *login* do Zabbix a figura 29 aparecerá.

Figura 29 – Tela de *login* do Zabbix

Fonte: Do autor.

1.2 PROXY ZABBIX

A instalação do *proxy* também é bastante simples. Ela será realizada na máquina com endereço IP: 20.20.20.10.

aptitude install -y zabbix-proxy-mysql zabbix-get zabbix-sender

Da mesma forma que no servidor Zabbix, algumas informações serão solicitadas:

a) na tela *Configuring mysql-server-5.5*, digite uma senha para o administrador do banco dados e tecle *ENTER*, logo a seguir confirme a mesma senha;

b) em *Configuring zabbix-proxy-mysql*, escolha *Yes* para que o Zabbix crie o banco de dados automaticamente;
c) a seguir digite uma senha do usuário *root* e uma para o usuário *zabbix* do banco de dados e confirme-a em seguida;

No arquivo de configurações, altere as informações *Server* e *Hostname* como no exemplo, e após reinicie o serviço:

```
vim /etc/zabbix/zabbix_proxy.conf
---------------------------------
Server=10.10.10.10                  # ip do servidor Zabbix
Hostname=zabbixproxyfilial
LogFile=/var/log/zabbix/zabbix_proxy.log
LogFileSize=0
PidFile=/var/run/zabbix/zabbix_proxy.pid
DBHost=localhost
DBName=zabbix_proxy
DBUser=zabbix
DBPassword=*******
DBSocket=/var/run/mysqld/mysqld.sock
ExternalScripts=/usr/lib/zabbix/externalscripts
FpingLocation=/usr/bin/fping
Fping6Location=/usr/bin/fping6
---------------------------------

/etc/init.d/zabbix-proxy restart
```

1.3 AGENTE ZABBIX

A instalação do agente será feita em todas as máquinas da infraestrutura:

```
aptitude install -y zabbix-agent zabbix-sender
```

O agente Zabbix possui configurações diferentes para cada máquina:

```
[Matriz - Servidor Zabbix]
vim /etc/zabbix/zabbix_agentd.conf
-----------------------------------------------
PidFile=/var/run/zabbix/zabbix_agentd.pid
LogFile=/var/log/zabbix/zabbix_agentd.log
LogFileSize=0
Server=127.0.0.1
ServerActive=127.0.0.1
Hostname=zabbixservermatriz
Include=/etc/zabbix/zabbix_agentd.d/
-----------------------------------------------
```

[Matriz – Servidor de Banco de Dados]
vim /etc/zabbix/zabbix_agentd.conf

PidFile=/var/run/zabbix/zabbix_agentd.pid
LogFile=/var/log/zabbix/zabbix_agentd.log
LogFileSize=0
Server=10.10.10.10
ServerActive=10.10.10.10
Hostname=dbservermatriz
Include=/etc/zabbix/zabbix_agentd.d/

[Matriz – Servidor HTTP]
vim /etc/zabbix/zabbix_agentd.conf

PidFile=/var/run/zabbix/zabbix_agentd.pid
LogFile=/var/log/zabbix/zabbix_agentd.log
LogFileSize=0
Server=10.10.10.10
ServerActive=10.10.10.10
Hostname=webservermatriz
Include=/etc/zabbix/zabbix_agentd.d/

[Matriz – Servidor E-mail]
vim /etc/zabbix/zabbix_agentd.conf

PidFile=/var/run/zabbix/zabbix_agentd.pid
LogFile=/var/log/zabbix/zabbix_agentd.log
LogFileSize=0
Server=10.10.10.10
ServerActive=10.10.10.10
Hostname=mxservermatriz
Include=/etc/zabbix/zabbix_agentd.d/

[Matriz – Firewall]
vim /etc/zabbix/zabbix_agentd.conf

PidFile=/var/run/zabbix/zabbix_agentd.pid
LogFile=/var/log/zabbix/zabbix_agentd.log
LogFileSize=0
Server=10.10.10.10
ServerActive=10.10.10.10
Hostname=firewallmatriz
Include=/etc/zabbix/zabbix_agentd.d/

[Filial – Zabbix Proxy]
vim /etc/zabbix/zabbix_agentd.conf

PidFile=/var/run/zabbix/zabbix_agentd.pid
LogFile=/var/log/zabbix/zabbix_agentd.log
LogFileSize=0
Server=20.20.20.10
ServerActive=20.20.20.10
Hostname=zabbixproxyfilial
Include=/etc/zabbix/zabbix_agentd.d/

[Filial – Servidor de Banco de Dados]
vim /etc/zabbix/zabbix_agentd.conf

PidFile=/var/run/zabbix/zabbix_agentd.pid
LogFile=/var/log/zabbix/zabbix_agentd.log
LogFileSize=0
Server=20.20.20.10
ServerActive=20.20.20.10
Hostname=dbserverfilial
Include=/etc/zabbix/zabbix_agentd.d/

[Filial – Servidor HTTP]
vim /etc/zabbix/zabbix_agentd.conf

PidFile=/var/run/zabbix/zabbix_agentd.pid
LogFile=/var/log/zabbix/zabbix_agentd.log
LogFileSize=0
Server=20.20.20.10
ServerActive=20.20.20.10
Hostname=webserverfilial
Include=/etc/zabbix/zabbix_agentd.d/

[Filial – Servidor FTP]
vim /etc/zabbix/zabbix_agentd.conf

PidFile=/var/run/zabbix/zabbix_agentd.pid
LogFile=/var/log/zabbix/zabbix_agentd.log
LogFileSize=0
Server=20.20.20.10
ServerActive=20.20.20.10
Hostname=ftpserverfilial
Include=/etc/zabbix/zabbix_agentd.d/

[Filial – Firewall]
vim /etc/zabbix/zabbix_agentd.conf

PidFile=/var/run/zabbix/zabbix_agentd.pid
LogFile=/var/log/zabbix/zabbix_agentd.log
LogFileSize=0
Server=20.20.20.10
ServerActive=20.20.20.10
Hostname=firewallfilial
Include=/etc/zabbix/zabbix_agentd.d/

Após alterar cada arquivo é necessário reiniciar o serviço *zabbix_agent,* para que as alterações comecem a funcionar, com o comando:

/etc/init.d/zabbix_agentd restart

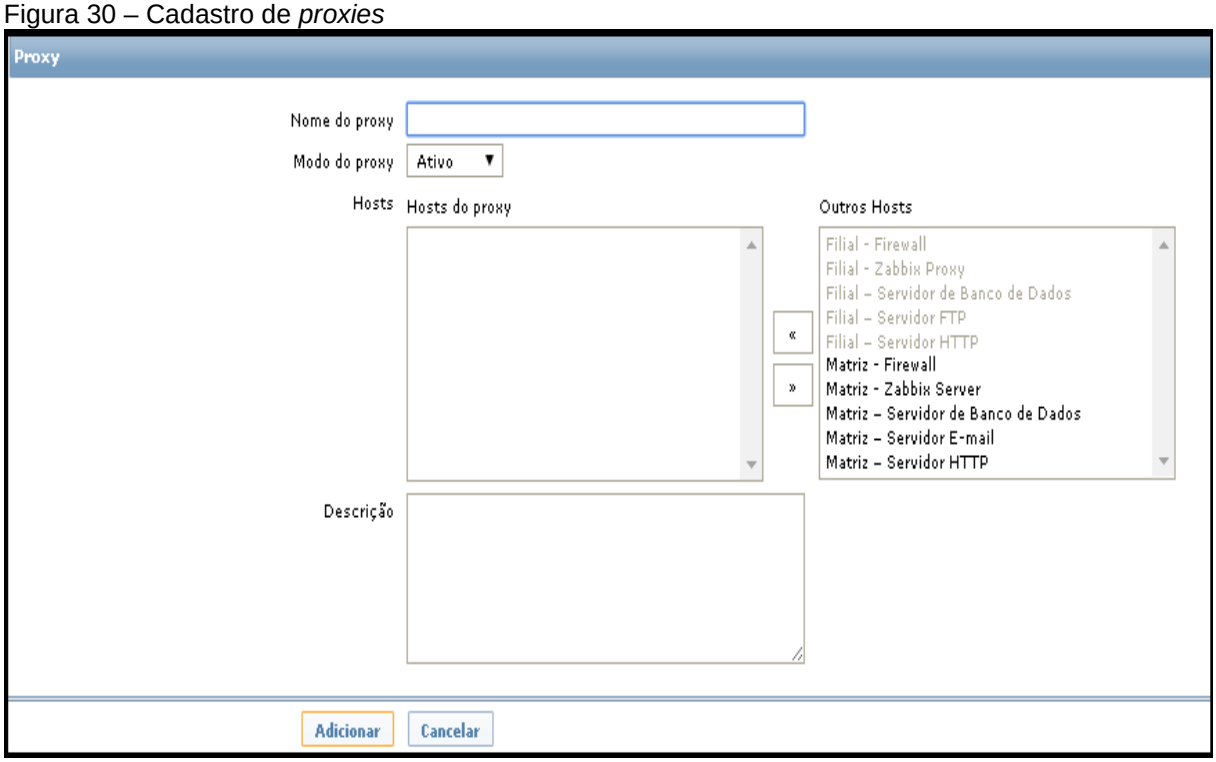
1.4 MONITORAMENTO

O primeiro passo para iniciar a configuração dos monitoramentos, consiste em acessar a interface Web do Zabbix com um navegador no endereço: *http://10.10.10.10/zabbix*. O usuário padrão é *admin* sua senha é *zabbix* (letras minúsculas). Após o *login* será feita a troca do idioma na interface Web, clicar em *"profile"* (em cima a direita), na guia *"user",* em *"Language"* escolher *"Portuguese (PT_BR)"* e clicar em *"Update"*.

1.4.1 Cadastro de proxies

Antes de incluir os servidores é necessário cadastrar os *proxies* que farão parte da infraestrutura. Os servidores na filial precisam do *proxy* em seu cadastro para ser monitorados, sem ele não é possível coletar dados. Para cadastra-lo é preciso acessar o menu: "*Administração"*, *"Proxies"* e clicar no botão "*Criar proxy"*. Preencher os campos *"Nome do proxy"* para "*zabbixproxyfilial"* e o *"Modo do proxy"* para *"Ativo"*, figura 30.

Figura 30 – Cadastro de *proxies*



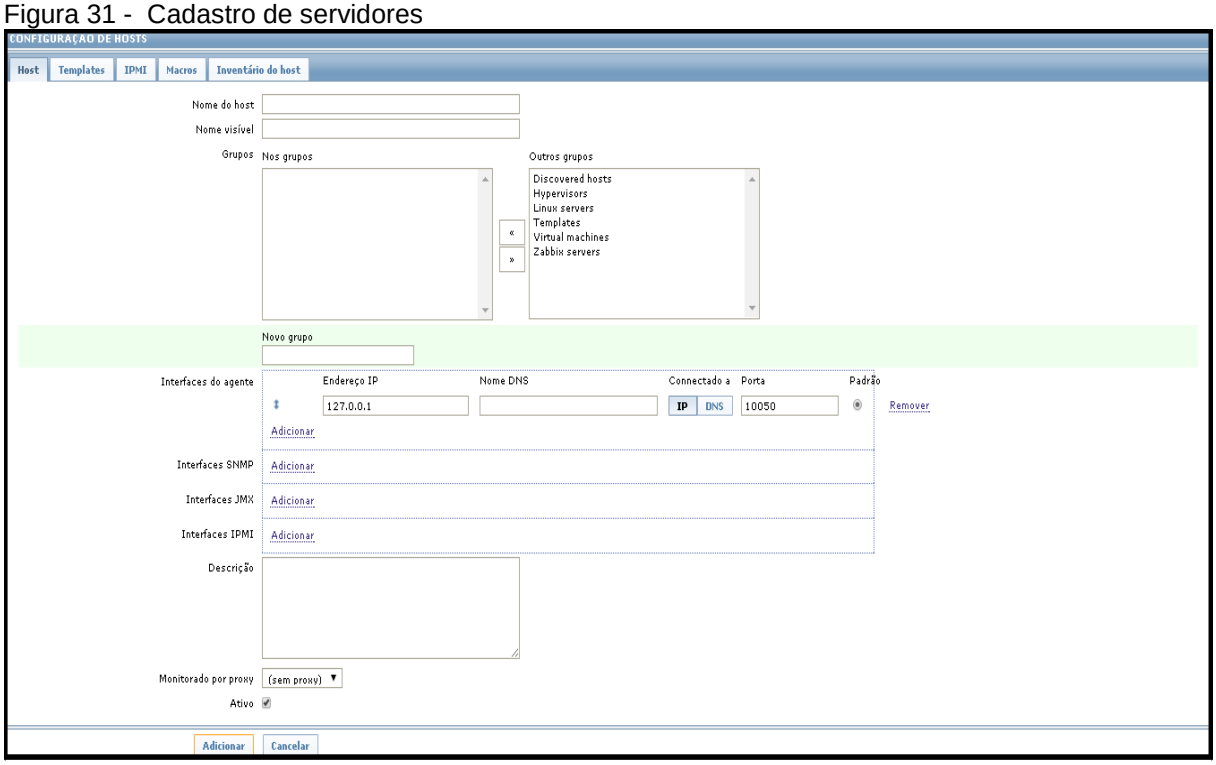
Fonte: Do autor.

1.4.2 Cadastro de servidores

O Zabbix possui um sistema de *templates* com os aplicativos e os itens que podem ser monitorados. Basta adicionar, cada um relacionado, ao item que se quer monitorar no servidor para que o processo comece a funcionar.

Para cadastrar um servidor novo basta clicar no *menu*: *Configurações, Hosts, Criar host (botão),* a janela de cadastro será exibida, figura 31.

Figura 31 - Cadastro de servidores



Fonte: Do autor.

Para cadastrarmos o monitoramento do servidor Zabbix, incluiremos os dados na guia *Host*:

Matriz – Zabbix Server
[Host]
Nome do host: zabbixservermatriz
Nome visível: Matriz – Zabbix Server
Grupos: Linux Server e Zabbix server
Endereço IP: 127.0.0.1
Monitorado por proxy: (sem proxy)
Ativo: marcar campo

Na guia *templates*, figura 31. Adicionaremos os itens: *Template App Zabbix Server*, *Template OS Linux e Template App MySQL*. Basta digitar o inicio da

palavra que aparecerão sugestões para escolha. Para adicioná-los efetivamente é necessário clicar em *Adicionar* logo abaixo do campo e para finalizar clicar no botão *Adicionar*.

Figura 32 – Vinculação de *templates* aos *hosts*

Fonte: Do autor.

Para os outros servidores utilizar os dados:

Matriz – Servidor de Banco de Dados
[Host]
Nome do host: dbservermatriz
Nome visível: Matriz – Servidor de Banco de Dados
Grupos: Linux Server
Endereço IP: 10.10.10.11
Monitorado por proxy: (sem proxy)
Ativo: marcar campo
[Templates]
Template OS Linux, Template App MySQL, Template ICMP Ping

Matriz – Servidor HTTP
[Host]
Nome do host: webservermatriz
Nome visível: Matriz – Servidor HTTP
Grupos: Linux Server
Endereço IP: 10.10.10.12
Monitorado por proxy: (sem proxy)
Ativo: marcar campo
[Templates]
Template OS Linux, Template ICMP Ping, Template App HTTP Service

Matriz – Servidor E-mail
[Host]
Nome do host: mxservermatriz
Nome visível: Matriz – Servidor E-mail
Grupos: Linux Server
Endereço IP: 10.10.10.13
Monitorado por proxy: (sem proxy)
Ativo: marcar campo
[Templates]
Template OS Linux, Template ICMP Ping, Template App IMAP Service, Template App POP Service, Template App SMTP Service

Matriz – Firewall
[Host]
Nome do host: firewallmatriz
Nome visível: Matriz – Firewall
Grupos: Linux Server
Endereço IP: 10.10.10.1
Monitorado por proxy: (sem proxy)
Ativo: marcar campo
[Templates]
Template OS Linux, Template ICMP Ping

Filial – Zabbix Proxy
[Host]
Nome do host: zabbixproxyfilial
Nome visível: Filial – Zabbix Proxy
Grupos: Linux Server
Endereço IP: 20.20.20.10
Monitorado por proxy: (sem proxy)
Ativo: marcar campo
[Templates]
Template OS Linux, Template ICMP Ping, Template App Zabbix Proxy

Filial – Servidor de Banco de Dados
[Host]
Nome do host: dbserverfilial
Nome visível: Filial – Servidor de Banco de Dados
Grupos: Linux Server
Endereço IP: 20.20.20.11
Monitorado por proxy: zabbixproxyfilial
Ativo: marcar campo
[Templates]
Template OS Linux, Template App MySQL, Template ICMP Ping

Filial – Servidor HTTP
[Host]
Nome do host: webserverfilial
Nome visível: Filial – Servidor HTTP
Grupos: Linux Server
Endereço IP: 20.20.20.12
Monitorado por proxy: zabbixproxyfilial
Ativo: marcar campo
[Templates]
Template OS Linux, Template ICMP Ping, Template App HTTP Service

Filial – Servidor FTP
[Host]
Nome do host: ftpserverfilial
Nome visível: Filial – Servidor FTP
Grupos: Linux Server
Endereço IP: 20.20.20.13
Monitorado por proxy: zabbixproxyfilial
[Templates]
Template OS Linux, Template ICMP Ping, Template App FTP Service

Filial – Firewall
[Host]
Nome do host: firewallfilial
Nome visível: Filial – Firewall
Grupos: Linux Server
Endereço IP: 20.20.20.1

Monitorado por proxy: zabbixproxyfilial
Ativo: marcar campo
[Templates]
Template OS Linux

Para o monitoramento do banco *MySQL* é necessário a criação de dois arquivos nas máquinas monitoradas, para que o servidor Zabbix acesse as informações necessárias. Um arquivo com o usuário e senha de e outro com as ações do monitoramento, exemplos a seguir:

vim /etc/zabbix/.my.cnf

[mysql]
user=zabbix
password=senha

[mysqladmin]
user=zabbix
password=senha
---

vim /etc/zabbix/zabbix_agentd.d/userparameter_mysql.conf

UserParameter=mysql.status[*],echo "show global status where Variable_name='$1';" | HOME=/etc/zabbix mysql -N | awk '{print $$2}'
UserParameter=mysql.size[*],echo "select sum($(case "$3" in both|"") echo "data_length+index_length";; data|index) echo "$3_length";; free) echo "data_free";; esac)) from information_schema.tables$([["$1" = "all" || ! "$1"]] || echo " where table_schema='$1'")$([["$2" = "all" || ! "$2"]] || echo "and table_name='$2'");" | HOME=/etc/zabbix mysql -N

UserParameter=mysql.ping,HOME=/etc/zabbix mysqladmin ping | grep -c alive
UserParameter=mysql.version,mysql -V

O resultado do cadastro de servidores, figura 33, apresentará além dos servidores, as aplicações, os itens, *templates* e demais de informações.

Figura 33 – Servidores cadastrados

Fonte: Do autor.

1.4.3 Notificações

Para a notificação por e-mail é necessário instalar um servidor SMTP e configurá-lo para repassar as mensagens. O servidor escolhido foi o *Postfix*, e será utilizada uma conta de e-mail gratuita para envio de alertas:

[Instalação]
aptitude install -y postfix bsd-mailx

[Configuração]
postconf -e "alias_maps = hash:/etc/aliases"

vim /etc/postfix/aliases
----
root: zabbix
----

newaliases

[Configuração da conta: Google]
postconf -e "relayhost = [smtp.gmail.com]:587"
postconf -e "smtp_sasl_auth_enable = yes"
postconf -e "smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd"
postconf -e "smtp_sasl_security_options = noanonymous"
postconf -e "smtp_tls_CAfile = /etc/postfix/cacert.pem"
postconf -e "smtp_use_tls = yes"

vim /etc/postfix/sasl_passwd

[smtp.gmail.com]:587 seu_e-mail@gmail.com:sua_senha

[Permissões]
chmod 600 /etc/postfix/sasl_passwd

postmap /etc/postfix/sasl_passwd

[Validação do Certificado SSL]
cat /etc/ssl/certs/Thawte_Premium_Server_CA.pem | tee -a /etc/postfix/cacert.pem

[Reinicialização do serviço]
/etc/init.d/postfix reload

[Teste de envio]
date | mail -s "teste `hostname`" email@empresa.com.br

Na interface do Zabbix procede-se o cadastro de uma nova mídia em *Administração*, *Tipos de mídias*, clicar em *Criar tipo de mídia* e preencher os campos conforme a figura 34:

Figura 34 – Tipo de mídia para envio de notificação

Fonte: Do autor.

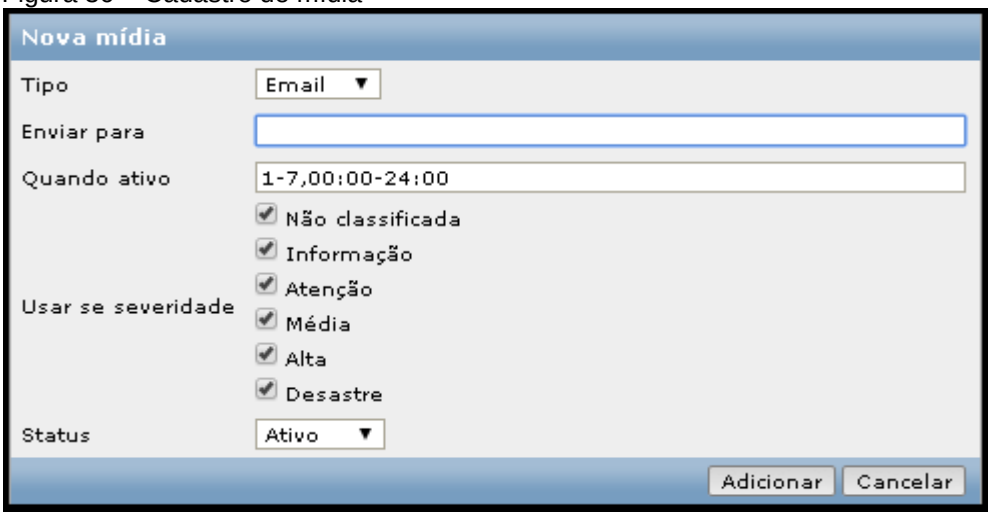
Ainda é necessário atribuir a nova mídia criada ao usuário que receberá as notificações. Em *Administração*, *Usuários*, alterar o *combobox* (em cima, à direita), de *Grupo de usuários*, para *Usuários*. Os usuários cadastrados serão apresentados, edite o usuário *admin*, clicando sobre ele e vá até a guia *Mídias*, figura 35.

Figura 35 – Configuração de mídias

Fonte: Do autor.

No cadastro para a nova mídia é que acrescentamos o e-mail que receberá os alertas, além de configurarmos o período de envio e a severidade do evento notificada, figura 36. Escolhe-se o *Tipo de mídia*: *E-mail*, *Enviar para: seu_e-mail@empresa*, depois é só confirmar clicando no botão *Adicionar* e finalizado no botão *Atualizar* da figura 35.

Figura 36 – Cadastro de mídia



Fonte: Do autor.

Por último a configuração das ações relacionadas com o envio de notificações. Acesse o *menu Configurações, Ações* e clique em *Criar ação*. Na guia *Ação*, preencher o campo *Nome* com *Zabbix Notificações* e marcar o campo *Mensagem da recuperação* para que um e-mail também seja enviado após a recuperação da falha. Na guia *Condições* preencha os campos: *Tipo do cálculo* com *E/OU*. Apague se houve condições antigas e adicione as seguintes: *Severidade da trigger = Desastre, Severidade da trigger = Alta, Severidade da trigger = Média, Severidade da trigger = Atenção, Severidade da trigger = Informação*. Na guia *Ações*, adicione uma nova e preencha os campos com: *Tipo da operação: Enviar mensagem*, *Enviar para usuários:* adicionar o usuário *Admin (Zabbix Administrator)* e marque o campo *Mensagem padrão*. Clique em *Adicionar* para finalizar.

1.4.4 Comandos remotos

São utilizados para executar um comando ou programa no servidor, após um evento ocorrer. Permitem que serviços sejam reiniciados quando o monitoramento detecta que pararam de responder. O exemplo a seguir foi utilizado com um servidor de e-mail, reiniciando os serviços POP, IMAP, SMTP quando estiverem parados.

Primeiro é necessário alterar o arquivo de configuração do agente Zabbix para que aceite a execução de comandos remotos, após será necessário instalar o software *sudo* e configurara quais comandos poderão ser executados sem a necessidade de *login* no sistema operacional. Depois é necessário atribuir o *Shell Bash* para o usuário *zabbix* (por padrão não possui nenhum), somente assim ele poderá executar o comando:

[agente]
vim /etc/zabbix/zabbix_agentd.conf

EnableRemoteCommands=1
LogRemoteCommands=1

[sudo]
aptitude install sudo

visudo

#Servidor SMTP
zabbix ALL=(ALL) NOPASSWD: /etc/init.d/postfix restart
#Servidor POP/IMAP
zabbix ALL=(ALL) NOPASSWD: /etc/init.d/dovecot restart

[Shell]
vipw
----
zabbix:x:108:114::/var/lib/zabbix/:/bin/bash

Faltando somente atribuir as ações para a execução dos comandos para concluir. Acesse o *menu Configurações, Ações* e clique em *Criar ação*. Na guia *Ação*, preencher o campo *Nome* com *{HOSTNAME} DAEMON DOVECOT OFF*. Na guia *Condições* preencha os campos: *Tipo do cálculo* com *E/OU*. Apague se houve condições antigas e adicione as seguintes: *Status de manutenção não em manutenção, Valor da trigger = INCIDENTE, Trigger = Matriz - mxserver: POP service is down on Matriz – mxserver, Trigger = Matriz - mxserver: IMAP service is down on Matriz – mxserver*. Na guia *Ações*, adicione uma nova e preencha os campos *Tipo da operação com Comando remoto*, em *Lista alvo* adicionar o servidor que o comando será executado, Tipo escolher Script personalizado, *Executar em* escolher *Agent Zabbix* e em Comandos: *sudo /etc/init.d/dovecot restart*. Criar uma nova ação para cada serviço que o Zabbix reiniciará.

APÊNDICE C – TESTES

1 TESTE DE CPU

Script feito para o Shell Bash, utilizado para os testes de CPU:

```
vim teste_cpu.sh
-----------------------
#!/bin/bash

I="0"
NTESTES="10"
TINTERVALO="300"      #tempo em segundos
TTESTE="300s"

while [ $I -lt $NTESTES ]
do
  stress --cpu 8 --io 4 --vm 2 --vm-bytes 128M --timeout "$TTESTE"
  sleep $TINTERVALO
  I=$[$I+1]
done
-----------------------

[Torna-lo executável]
chmod +x teste_cpu.sh

[Execução]
./ teste_cpu.sh
```

2 TESTE DE MEMÓRIA

Configuração para o Tomcat utilizar o mínimo de 128Mb de memória RAM.

```
vim /etc/default/tomcat
------------------------------
JAVA_OPTS="-Xms128m -Xmx256m"
------------------------------

/etc/init.d/tomcat restart
```

3 TESTE DE CHECAGEM WEB

Comando para mover o arquivo HTML do diretório público.

```
mv /var/www/html/index.html /var/www/
```

4 TESTE DE REDE

Instalação e configuração do *proxy squid* para limitação da banda de internet.

[Instalação]
aptitude install -y squid3

[Configuração]
vim /etc/squid3/squid.conf
---
acl all src
acl localhost src 127.0.0.1
acl redelocal src 10.0.0.0/8
#acl redelocal src 172.16.0.0/12
#acl redelocal src 192.168.0.0/16
acl SSL_ports port 443 563 873
acl Safe_ports port 80 21 443 70 210 1025-65535 280 488 591 777 631 873 901
acl purge method PURGE
acl CONNECT method CONNECT

acl controlebanda src redelocal
delay_pools 1
delay_class 1 2
delay_parameters 1 128000/128000 128000/128000 #128kb = 1Mbps
delay_access 1 allow controlebanda

http_access allow manager localhost
http_access deny manager
http_access allow purge localhost
http_access deny purge
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports
http_access allow localhost
http_access allow redelocal
http_access allow all
icp_access allow redelocal
icp_access deny all
http_port 10.0.0.1:3128 transparent
#hierarchy_stoplist cgi-bin ?
access_log /var/log/squid3/access.log squid
refresh_pattern ^ftp: 1440 20% 10080
refresh_pattern ^gopher: 1440 0% 1440
refresh_pattern -i (/cgi-bin/|\?) 0 0% 0
refresh_pattern (Release|Packages(.gz))$ 0 20% 2880*
refresh_pattern .
hosts_file /etc/hosts
coredump_dir /var/spool/squid3
cache_effective_user proxy
cache_effective_group proxy
cache_mem 512 MB
cache_dir diskd /var/spool/squid3/1 10240 16 256
cache_dir diskd /var/spool/squid3/2 10240 16 256
cache_dir diskd /var/spool/squid3/3 10240 16 256
maximum_object_size 250 MB

[Criação do cache]
/etc/init.d/squid3 stop
squid3 -f /etc/squid3/squid.conf -z

[Inicialização do serviço]
/etc/init.d/squid3 start

[Configurações do firewall]
iptables -t nat -N proxy
iptables -t nat -A PREROUTING -s 10.10.10.0/24 -i eth1 -p tcp -m multiport --dports 80,443 -j proxy
iptables -t nat -A proxy -p tcp -j DNAT --to 10.10.10.1:3128
iptables -t nat -I proxy -p tcp -m tcp --dport 443 -j ACCEPT

/etc/init.d/netfilter-persistent save

APÊNDICE D – ARTIGO

Simulação de Utilização da Ferramenta de Gerência de Redes Zabbix para Monitoramento de Ativos e Análise de Tendências no Auxílio à Administração de Ambientes Distribuídos

Sivaldo Martins[1], Paulo João Martins[2]

[1]Acadêmico do Curso de Ciência da Computação – Unidade Acadêmica de Ciências, Engenharias e Tecnologias – Universidade do Extremo Sul Catarinense (UNESC) –Criciúma – SC – Brazil

[2]Professor do Curso de Ciência da Computação – Unidade Acadêmica de Ciências, Engenharias e Tecnologias – Universidade do Extremo Sul Catarinense (UNESC) – Criciúma – SC – Brazil

sivasmartins@gmail.com, pjm@unesc.net

Abstract. *This project studied and applied the use of the Zabbix as administrator support tool in the management of distributed networks, by fault detection and behaviors that generate unanticipated problems. The study was fulfilled in a controlled environment designed to tests with simulation of network errors and events. The effectiveness of the tool in supporting decisions made by the administrator in the management application concepts could be verified. During the study, we searched to understand the environment and the tool to obtain a wider knowledge during the implementation. The tests have shown that the application has many functions which can assist the administrator in charge of managing distributed networks.*

Resumo. *Este projeto estudou e aplicou o uso do Zabbix como ferramenta de apoio ao administrador no gerenciamento de redes distribuídas, por meio da detecção de falhas e comportamentos não previstos capazes de gerar problemas. O estudo foi realizado em um ambiente controlado idealizado para testes com simulação de erros e eventos de rede, podendo-se então verificar a eficácia da ferramenta no apoio às decisões tomadas pelo administrador na aplicação dos conceitos de gerência. Durante o estudo procurou-se entender o ambiente e a ferramenta com o intuito de se obter um conhecimento mais amplo na implementação. Os testes demonstraram que a aplicação possui muitas funções capazes de auxiliar o administrador na função de gerenciar redes distribuídas.*

1. Introdução

Manter os recursos funcionando em um ambiente de redes é uma tarefa complexa, demandando tempo para uma análise detalhada sobre as ações que devem ser executadas na solução dos problemas encontrados [Comer 2007].

O uso de ferramentas, para a divisão da complexidade de gerenciar uma rede local ou distribuída, passa a ser relevante para quem busca eficiência, seja na ação reativa em caso de o problema ter ocorrido, ou na abordagem proativa, antes da ocorrência [Stallings 2005].

Existem muitas ferramentas capazes de auxiliar o administrador nessa tarefa. O uso do Zabbix deu-se pela sua eficiência e flexibilidade, funções diversificadas de monitoramento de desempenho, serviços de rede e conteúdo ofertado, opções de notificação, apresentação dos dados coletados na forma de relatórios e gráficos que auxiliam na identificação de eventos e erros que possam levar ao mau funcionamento do ambiente, corrigindo-os antes que o desempenho seja afetado [Horst, Pires e Déo 2015].

2. Gerência de Redes

Gerenciar redes engloba os aspectos de configuração, controle e relatório, úteis para entender o seu funcionamento, servindo de auxílio na tomada de decisão durante uma manutenção [Farrel 2005].

Somente pelo esforço humano é inviável o gerenciamento do ambiente formado pelas redes de uma corporação. A descentralização dos serviços, os ambientes distribuídos e a dispersão dos recursos contribuem para a complexidade, por isso o emprego de ferramentas que automatizem os procedimentos de monitoramento e o controle sobre os itens, tornam-se essenciais [Stallings 2005].

Elas irão monitorar o funcionamento e notificar sempre que houver algum evento incomum, fazendo consultas nos ativos e nos serviços de rede para determinar se estão operando corretamente [Casad e Willsey 1999].

3. Arquitetura de Gerência TCP/IP

O modelo utilizado no gerenciamento de redes TCP/IP é composto pela entidade gerenciadora responsável pela aquisição, análise e processamento das informações, pelo agente que coleta os dados, pela base de informações de gerenciamento associada aos objetos gerenciados e o protocolo de gerenciamento utilizado para a comunicação [Kurose e Ross 2006].

4. Ferramentas de Gerenciamento

Existem várias ferramentas que podem ser usadas para gerenciar redes, apresentando maior ou menor grau de complexidade e abrangência na análise.

As ferramentas de rede analisam os problemas de conexão em rede, os analisadores de protocolos servem para inspecionar os pacotes de um protocolo, os sistemas de gerenciamento de redes analisam o desempenho e o status dos componentes da rede em busca de problemas, são compostos por ferramentas para monitoramento e controle, espalhados entre os nós da rede [Junior et al 1999].

5. Implantação da Ferramenta Zabbix no Gerenciamento de Ambientes Distribuídos

Este estudo teve como objetivo implantar a ferramenta Zabbix testando e avaliando os resultados de sua eficácia, na detecção de erros e situações de degradação. Foram monitorados e notificados quaisquer incidentes que ocorreram com os ativos e serviços de rede do ambiente simulado. As informações apuradas com o monitoramento foram utilizadas para geração de gráficos de desempenho, auxiliando a ação proativa do administrador, na resolução de problemas.

5.1. Modelagem do Ambiente

O ambiente proposto para o estudo é composto por um servidor físico que hospedou dez máquinas virtuais. Elas estão configuradas para formar duas redes conectadas via Internet,

simulando através de virtualização a matriz e a filial de uma empresa (figura 1). A infraestrutura de servidores possui alguns serviços configurados como o Post Office Protocol (POP), Simple Mail Transfer Protocol (SMTP), HyperText Transfer Protocol (HTTP), que serão monitorados durante os testes.

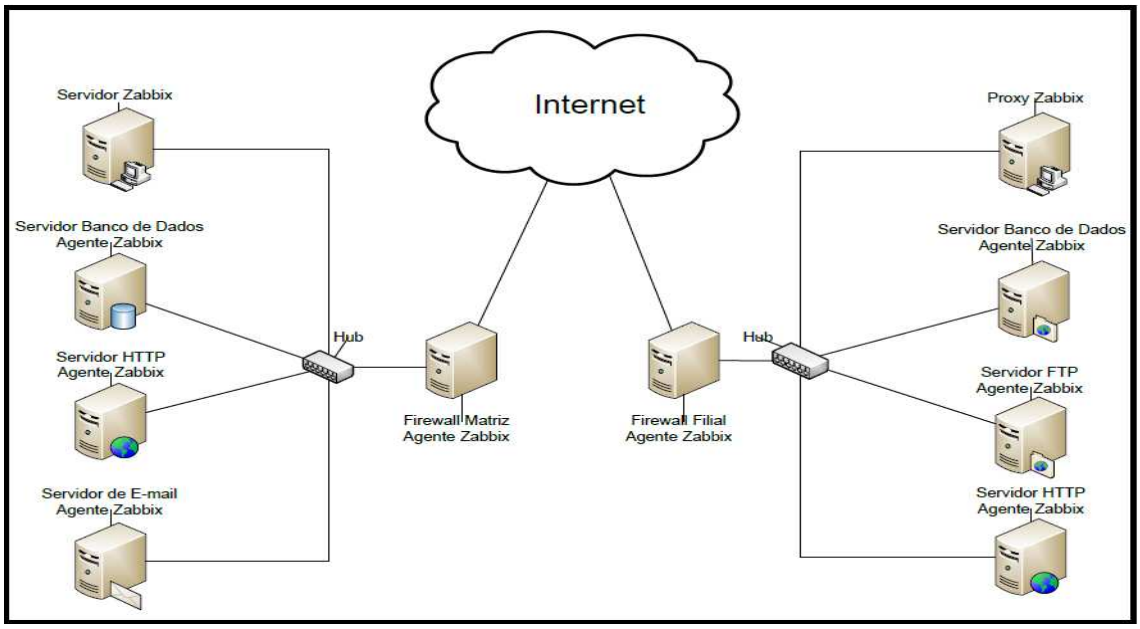


Figura 1. Estrutura do ambiente simulado.

5.2. Implantação do Ambiente

O sistema operacional hospedeiro utilizado no ambiente foi o Ubuntu 14.04, nele foi instalado o virtualizador Oracle VM VirtualBox 4.3.30. Para as máquinas virtuais foi utilizado o sistema operacional Debian GNU/Linux 8.2, instalado com um perfil mínimo, sem serviços desnecessários. Os serviços testados e a infraestrutura Zabbix foram instalados via repositório do sistema operacional para automatizar a resolução de dependências.

O gerente da infraestrutura Zabbix composto pelo servidor, sistema de banco de dados e a interface administrativa foi instalado na matriz da empresa, em uma única máquina virtual, por se tratar de um ambiente de testes pequeno e que não demanda alto desempenho. O *proxy* Zabbix foi instalado na filial, ele funcionará como um superagente armazenando os dados coletados pelos agentes e enviando-os ao servidor. Em caso de queda da comunicação, o envio se dará com o restabelecimento da mesma. O agente binário foi escolhido para a coleta dos dados pela facilidade de instalação e configuração. Sempre que um incidente é detectado e permanece ativo, é exibida uma notificação na interface gráfica, alertando e permitindo que seja feito o acompanhamento pelo responsável.

5.3. Resultados Obtidos

Nos testes realizados visando detectar gargalos de desempenho ocasionados pelo consumo excessivo de processamento, foi utilizado o software stress, disponível na árvore de pacotes do Debian GNU/Linux, ele executou testes em vários itens do sistema simulando uso excessivo dos componentes. Assim que o uso da *Central Process Unit* (CPU) aumentou o Zabbix fez a detecção e disparou a notificação sobre a ocorrência em ambos os servidores, na tela de incidentes (figura 2).

Os tempos de resposta para o ambiente remoto e o local diferem, pois os dados remotos passam por uma camada extra na infraestrutura, o *proxy* Zabbix. A vantagem de

utilizá-lo em relação ao monitoramento direto é que, mesmo se houvesse perda da comunicação com o servidor, às informações ainda estariam armazenadas no banco de dados do *proxy*.

Figura 2. Notificação de uso intensivo do processador.

Analisando os gráficos de utilização de CPU, gerados pelos dados obtidos, é possível detectar que ocorrem picos de processamento nos servidores. Os gráficos de processos executando indicam ainda que durante esses picos de processamento o número de processos em execução aumenta consideravelmente (figura 3).

Figura 3. Gráficos de utilização de processador.

Os gráficos indicam que há processos com problemas, sendo necessária uma investigação mais detalhada nos servidores para solucionar o problema.

Há momentos aonde a degradação do ambiente vai ocorrendo de forma sutil com o passar do tempo, prevê-las é difícil e fundamental para evitar que causem a parada do sistema. A interpretação das informações coletadas pelos agentes por meio dos gráficos gerados é chamada de análise de tendências. O administrador, de posse dessas informações irá se antecipar a ocorrência de falhas.

O teste consistiu em adicionar arquivos ao disco do servidor, via transferência FTP, por um período de tempo até quase saturá-lo. Servidores de arquivos podem apresentar esse problema, principalmente se não são acessados diariamente pelo administrador para que sejam verificados. Por meio do gráfico gerado, foi possível perceber que o espaço livre disponível foi diminuindo com o passar do tempo (figura 4). Se o processo continuasse fatalmente ocorreria uma falha no servidor.

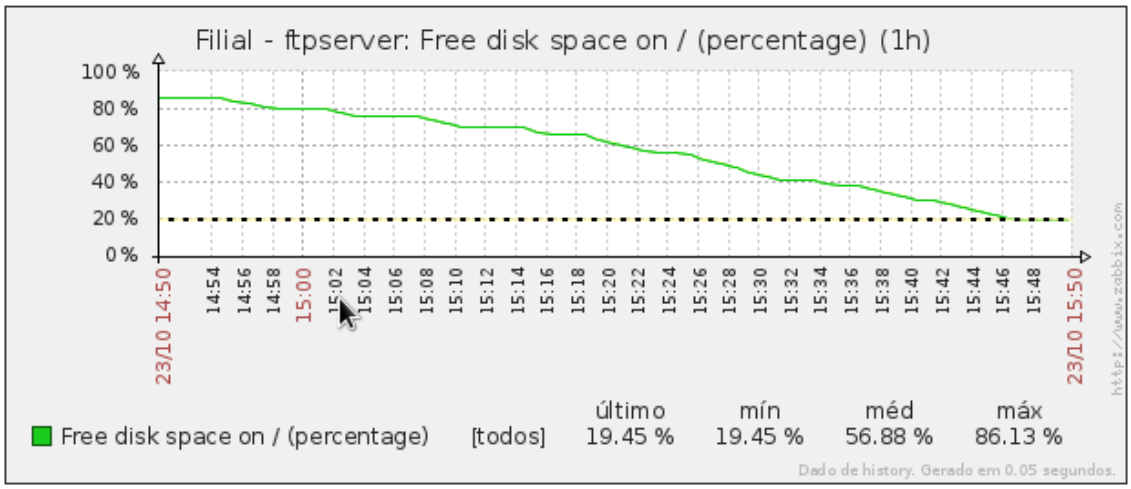


Figura 4. Gráfico de espaço livre em disco.

Neste caso a ferramenta, orientou o administrador a interferir no problema de falta de espaço em disco de forma proativa. O monitoramento de discos e partições do Zabbix, na configuração padrão ainda disparou uma notificação quando o espaço livre ficou menor que 20% do total. O administrador poderia configurar este valor conforme sua necessidade. A mesma análise poderia ser aplicada aos níveis de consumo de memória RAM ou links de Internet.

Normalmente em situações de falta de espaço é comum que um novo disco seja adicionado a estrutura do servidor. Fica para o administrador a tarefa de mapear e configurar esse novo disco, incluindo-o no sistema de gerenciamento de rede. Um diferencial do Zabbix é possuir um processo de descoberta interna automática, que pode ser configurado para descobrir discos, partições e interfaces de rede novas. Isso incide positivamente na redução de trabalho do administrador, modificando o perfil de monitoramento do servidor sem a necessidade de sua interferência.

Existem muitos serviços de rede sendo consumidos pelos usuários nas empresas. A parada desses serviços é um transtorno muito grande para os processos organizacionais. A checagem simples de portas, para verificar se o serviço está funcionando, é um item de monitoramento muito importante em uma infraestrutura.

Para esse teste, a checagem de portas foi executada em conjunto com a verificação de disponibilidade, coletando o retorno de pacotes ICMP em um servidor de e-mail executando os serviços POP, IMAP e SMTP. Na simulação a interface de rede do servidor foi desativada. Na notificação (figura 5), é possível ver o erro na resposta dos serviços e ainda notar o alerta de ausência de respostas para pacotes ICMP, indicação de que o servidor pode não estar ativo ou há algum possível problema de conectividade. A interpretação agregada dessas informações melhora o diagnóstico do problema pelo administrador.

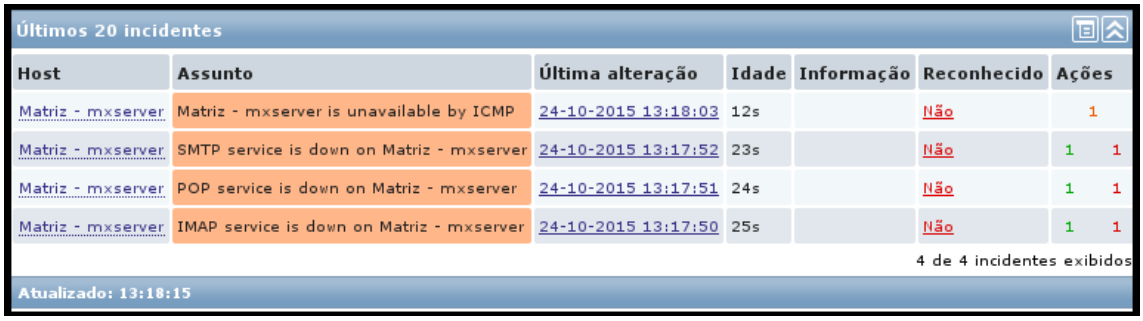


Figura 5. Notificação de falha em serviços.

O Zabbix ainda possui um sistema de ações integradas com os gatilhos de eventos. Quando eles ocorrem, ações podem ser configuradas para executar comandos remotos nos servidores, possibilitando reiniciar os serviços, caso o problema tivesse ocorrido somente nos serviços de rede, diminuindo o tempo de resposta para alguns incidentes.

Outra variação do teste de checagens simples foi feita em serviços Web. Quase toda organização usa algum tipo de serviço assim. O Zabbix monitorou além da porta do serviço, a disponibilidade e o desempenho do site, simulando a experiência de acesso do usuário. Esse diferencial, chamado de checagem Web, serve para detectar uma falha no site ao invés de um problema no serviço somente. A checagem Web busca alguma informação diferenciada no endereço monitorado do servidor, podendo ser um texto ou até mesmo a página resultante de uma autenticação no site.

Na notificação do teste (figura 6), a página padrão, criada durante a instalação do servidor HTTP, foi removida do diretório público e como resultado o alerta não indica problema no serviço, somente no conteúdo ofertado.

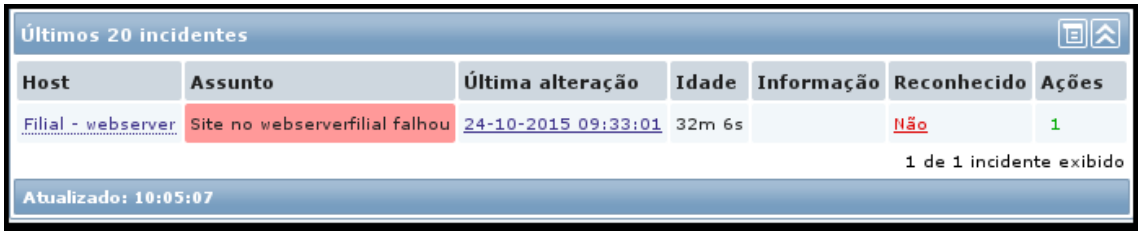


Figura 6. Notificação de checagem web.

Outro item que é muito exigido nos servidores é a memória. Alguns serviços acabam utilizando esse recurso ainda mais quando são muito requisitados, é o caso de servidores de aplicação e Web. Pode ainda haver erro no código de alguma aplicação que gere uma falha drenando esse recurso.

O teste consistiu em execução de um servidor de aplicação Java utilizando um valor mínimo de memória. Esse valor foi configurado para equivaler ao total de memória disponível, causando o consumo quase total desse recurso no servidor, assim que foi iniciado (figura 7).

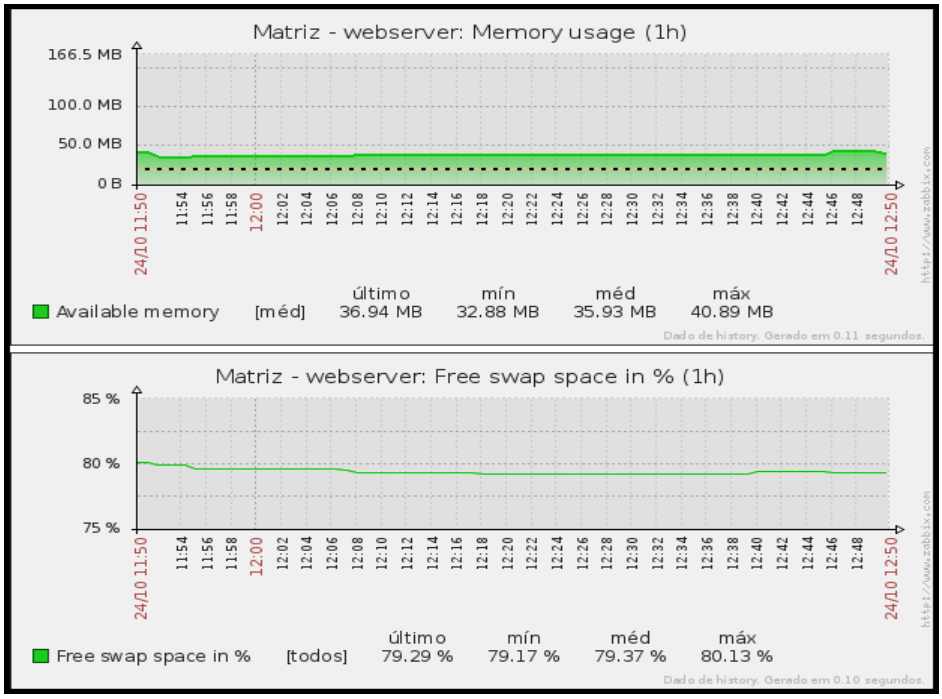


Figura 7. Gráficos de consumo de memória.

Com a diminuição da memória livre disponível, é possível observar que as operações de leitura e escrita no disco perderam desempenho. Analisando o gráfico de memória Swap livre, constata-se que o espaço de trocas também está sendo bem utilizado. Observando os dois gráficos pode-se pressupor que há correlação entre o baixo desempenho do disco e o autoconsumo das memórias. O gargalo pode estar sendo criado pelo processo de Swapping que a gerência de memória do sistema operacional está realizando.

Outro item problemático nas redes das empresas é o uso indiscriminado do link de Internet. Ações importantes na empresa podem deixar de ser feitas, por que alguém está fazendo uso indevido desse recurso.

O teste feito consistiu limitar artificialmente o link de Internet no firewall da matriz para 1 Mbps, para saturá-lo facilmente com a realização de um download. Através dos gráficos é possível observar o fluxo de informações atravessando o firewall.

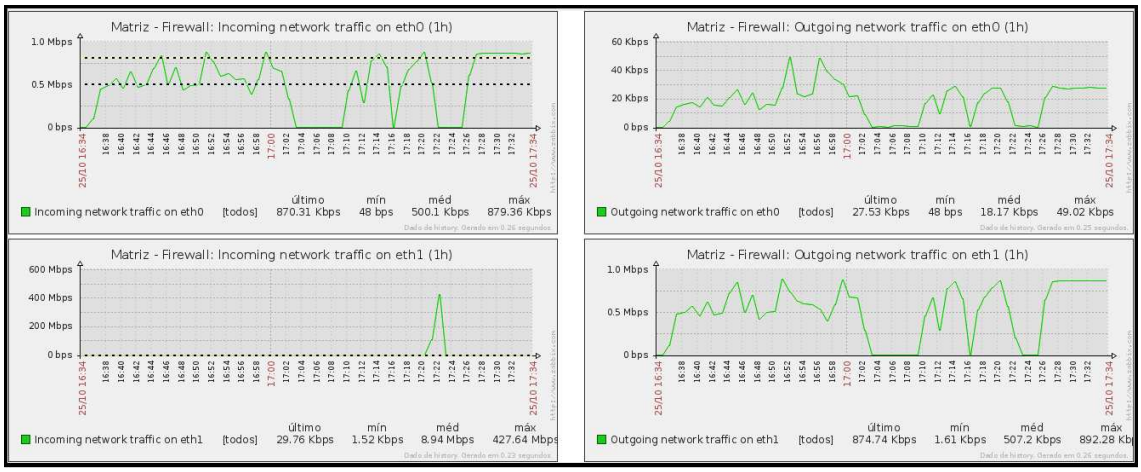


Figura 8. Consumo do link de Internet.

As informações dos gráficos de consumo, e a notificação configurada para alertar após o consumo maior que 50% e 80% do total (figura 9), servem para diminuir o tempo de resposta para uma ação corretiva no problema.

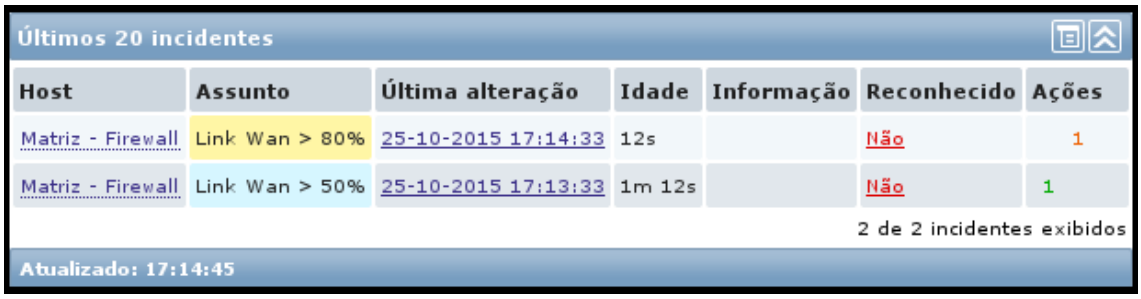


Figura 9. Alertas de consumo do link de Internet.

O Zabbix por padrão tem a capacidade de monitorar o banco de dados MySQL, sendo possível a geração de estatísticas sobre as operações de manipulação dos dados, além de poder observar a largura de banda dos dados recebidos e enviados nessas operações (figura 10).

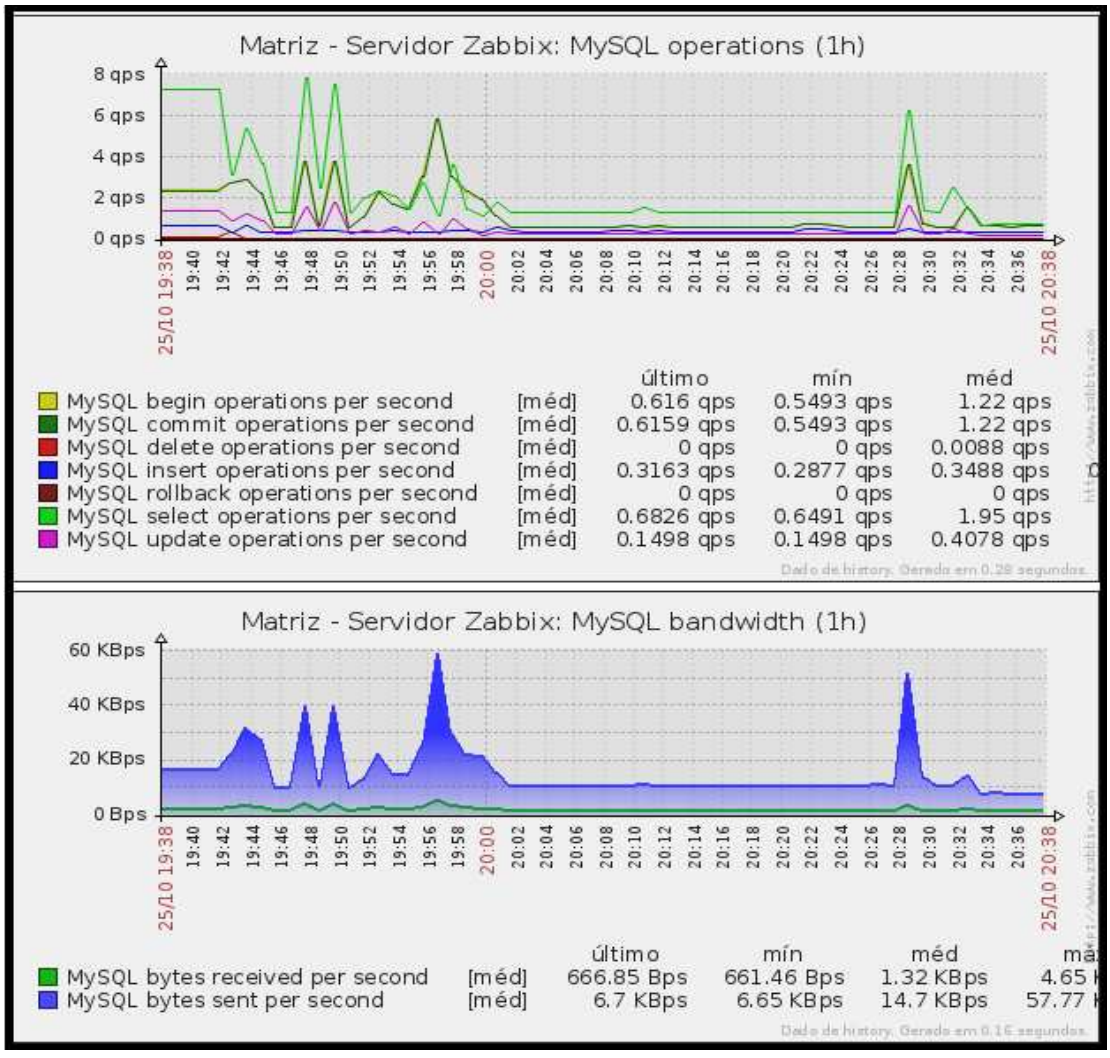


Figura 10. Monitoramento de banco de dados.

A ausência de alertas relacionados às operações no banco é um ponto negativo na ferramenta. O único alerta disponível é a verificação do serviço, para saber se ele está ativo.

Para melhorar a aplicação da ferramenta existem na Internet customizações desenvolvidas por usuários, que podem ser integradas ao Zabbix, para melhorar a experiência com o banco de dados. No site oficial, há documentação para que se criem as próprias notificações de forma flexível em seu ambiente, baseadas na coleta de informações feitas por plugins que o administrador poderá criar em qualquer linguagem, de acordo com sua necessidade.

A apresentação dos dados demonstra ser um diferencial na ferramenta. Ideal para a criação de centros de operação para controle da infraestrutura de rede. Pode-se customizar a exibição das informações com a criação de telas personalizadas agregando gráficos e mapas representando a infraestrutura de rede (figura 10).

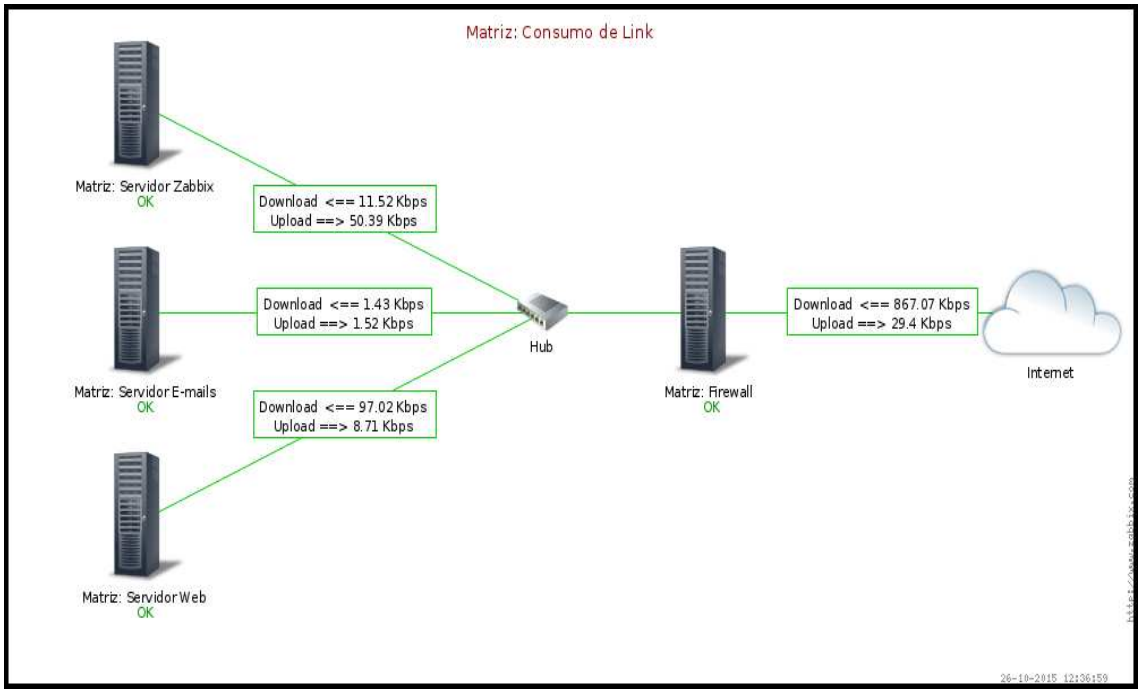


Figura 10. Mapa de rede.

O Zabbix demonstrou inúmeras funcionalidades para o monitoramento, alertas e relatórios. De itens de desempenho como opções de nível de utilização e consumo de recursos como CPU, memória, disco, rede, operações em banco de dados além de opções de notificação diversificadas, para garantir que o administrador seja avisado em caso de falhas.

O módulo administrativo foi projetado seguindo os conceitos das linguagens orientadas a objetos, permitindo que configurações feitas sejam reaproveitadas por meio de herança de propriedades, poupando tempo na configuração de novos itens.

Há centralização da autenticação via LDAP e controle apurado das ações dos usuários no sistema por auditoria.

O ponto falho encontrado é a comunicação não criptografada entre módulos, constituindo quebra de segurança nas comunicações pela Internet. Isso foi contornado utilizando uma VPN entre os firewalls.

Os testes aplicados procuraram demonstrar eventos reais em um ambiente simulado, e os benefícios obtidos pelo administrador no uso da ferramenta.

6. Conclusão

O uso de ferramentas de gerenciamento é cada vez mais necessário, já não sendo mais viável somente a intervenção manual no ambiente, frente à complexidade enfrentada pelo administrador. O Zabbix é uma dessas ferramentas capazes de auxiliar o administrador.

O estudo demonstrou sua capacidade na detecção de falhas e erros no monitoramento de ambientes, diferenciando-se por agregar testes, gerando melhores resultados. Adaptação dos perfis de monitoramento com a detecção automática dos novos itens. Ações proativas programadas, como a reinicialização dos serviços em caso de falha detectada, eficiência na antecipação de ações indesejáveis via gráficos de tendências de comportamento. O módulo *proxy* mostrou eficiência no auxílio a administração de ambientes distribuídos.

A apresentação das informações é diversificada, podendo ser customizada conforme as necessidades do administrador.

O ponto negativo encontrado é a ausência de criptografia na comunicação entre seus módulos, uma falha de segurança no monitoramento de redes pela Internet, sendo o problema superado utilizando-se uma VPN, configurada com o software OpenVPN, entre os firewalls da matriz e a filial.

Referências

Casad, J. e Willsey, B. (1999) “Aprenda em 24 horas Tcp/Ip”. Campus.

Comer, D. E. (2007) “Redes De Computadores e Internet”. Bookman.

Farrel, A. (2005) “A internet e seus Protocolos: uma análise comparativa”. Campus.

Horst, A. S., Pires, A. d. S. e Déo, A. L. B. (2015) “De A a ZABBIX”. Novatec.

Kurose, J. F. e Ross, K. W. (2006) “Redes de Computadores e a Internet: uma abordagem top-down”. Pearson Addison Wesley.

Stallings, W. (2005) “Redes e sistemas de comunicação de dados”. Elsevier.

Junior, J. H. T., Suavé, J. P., Moura, J. A. B. e Teixeira, S. d. Q. R. (1999) “Redes de Computadores: Serviços, Administração e Segurança”. Makron-Books.