Sergio Henrique Bento de Mira

Implantação de um sistema para monitoramento de rede e serviços

Lavras

2014

Sergio Henrique Bento de Mira

Implantação de um sistema para monitoramento de rede e serviços

Trabalho de Conclusão de Curso de Graduação apresentado ao Colegiado do Curso de Bacharelado em Ciência da Computação, para obtenção do título de Bacharel.

Universidade do Federal de Lavras – UFLA

Departamento de Ciência da Computação – DCC

Orientador: Neumar Costa Malheiros

Lavras

2014

SERGIO HENRIQUE BENTO DE MIRA

IMPLANTAÇÃO DE UM SISTEMA PARA MONITORAMENTO DE REDE E SERVIÇOS

Trabalho de Conclusão de Curso de Graduação apresentado ao Colegiado do Curso de Bacharelado em Ciência da Computação, para obtenção do título de Bacharel.

APROVADA em 26 de novembro de 2014.

Dr. Rêmulo Maia Alves

Dr. André Grützmann

Dr. Neumar Costa Malheiros (Orientador)

LAVRAS-MG
Novembro/2014

Este trabalho é dedicado à minha mãe,
Maria Raimunda Bento de Mira.

Agradecimentos

A Deus, por ter permitido que a superação se mostrasse sempre o caminho.

Aos meus pais Ernani de Mira (*in memorian*) e Maria Raimunda Bento de Mira, pelo amor, cuidado, confiança e apoio incondicional.

Ao meu irmão Jairo e ao meu amigo-primo Carlos Eduardo, por mostrarem-se sempre acima das dificuldades doando motivação sem precedentes.

A esta Universidade, seu corpo docente e administração que me deram a oportunidade de vislumbrar um horizonte antes inatingível, eivado pela acendrada confiança no mérito e ética nela presentes.

A Gerencianet Pagamentos do Brasil, seus gestores e meus supervisores, pela oportunidade, paciência, confiança e ambiente amistoso.

Ao meu orientador Professor Doutor Neumar Costa Malheiros, pela oportunidade, parceria, conselhos e suporte no pouco tempo e longa distância de convivências pessoal e profissional, pelas suas correções e incentivos.

Aos amigos e parceiros Davi, Bruno, Cristian, Matheus e Paulo, pelos tantos trabalhos e sofrimento em conjunto.

Aos amigos e companheiros diários de trabalho Alessandro, Eduardo, Marcos e Túlio, pela oportunidade de troca de conhecimento e convivências pessoal e profissional.

Às bandas *Rush*, *Dream Theater*, *Transatlantic*, *Pink Floyd*, *Kansas*, *King Crimson* e *Lynyrd Skynyrd*, por estimularem a curiosidade, a criatividade, a descoberta, a objetividade, a visão crítica e por suas reflexões sobre a vida em geral.

E a todos que, direta ou indiretamente, fizeram parte desta caminhada, o meu muito obrigado.

Resumo

Neste trabalho foi implantado na empresa Gerencianet Pagamentos do Brasil o Zabbix, um sistema para monitoramento de redes e serviços. Constam neste relatório algumas informações sobre a empresa, como sua história, mercado de atuação e produtos disponíveis neste mercado. Para a implantação do Zabbix, foi necessário conhecimento completo dos produtos da empresa, bem como do funcionamento interno da rede. Foram levantados e categorizados os ativos de rede a serem monitorados e apresentadas as características de configuração para o funcionamento do Zabbix e emissão de alertas aos administradores da infraestrutura da empresa. Com isso, pôde-se aplicar, diretamente, conceitos de Redes de Computadores, Sistemas Operacionais e conhecimentos em Linux, adquiridos ao longo do curso de Ciência da Computação.

Palavras-chaves: redes de computadores, sistemas operacionais e monitoramento de rede e serviços

Abstract

In this paper it was implemented Zabbix in the company Gerencianet Pagamentos do Brasil, a monitoring system for networks and services. This report contains some information about the company, like its history, active market and products available in this market. For deploying Zabbix, it was necessary to thorough knowledge of company products, as well as the inner workings of the network. Network assets to be monitored were surveyed and categorized and were displayed the characteristics of configuration for the operation of Zabbix and alerting company's infrastructure administrators. With that, it was possible apply concepts of Computer Networks, Operating Systems and Linux knowledge, acquired over the course of Computer Science.

Key-words: computer networks, operational systems and network and services monitoring

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

UFLA	Universidade Federal de Lavras
PRG	Pró-reitoria de Graduação
CEO	*Chief Executive Officer*
SEO	*Search Engine Optimization*
AS	*Autonomous System*
IP	*Internet Protocol*
DNS	*Domain Name System*
EUA	*Estados Unidos da América*
SMS	*Short Message Service*
SMTP	*Simple Mail Transfer Protocol*

Sumário

1 Introdução

O estágio é um conjunto de atividades de formação, pesquisa e prestação de serviços à comunidade e, por meio de aplicação de metodologias estudadas durante um curso de graduação, permite ao aluno a percepção dos conceitos vistos em sala de aula. Além disso, concede ao estagiário as experiências pessoal, profissional e técnica necessárias para tomada de decisões após a conclusão de seu curso.

Segundo a PRG/UFLA, o egresso do curso de Graduação em Ciência da Computação da UFLA é um profissional capaz de analisar situações cientificamente, identificar e resolver problemas, preocupar-se com atualização permanente de conhecimentos e de tomar decisões, com a finalidade de criar sistemas de software e hardware, sempre se atentando aos aspectos sociais, dentro dos princípios éticos. Tendo em vista este perfil de profissional, o estágio deve permitir ao aluno a atuação em diversas áreas, aumentando sua capacidade de solucionar problemas utilizando tecnologias atuais.

O documento em questão visa relatar as atividades executadas durante o estágio na empresa Gerencianet Pagamentos do Brasil, que envolveram a aplicação da área de Ciência da Computação, mais especificamente Sistemas Operacionais e Redes de Computadores, em um contexto bastante atual e muito importante no setor de Infraestrutura de qualquer empresa, o monitoramento de redes e de serviços que nelas são executados. O objetivo deste estágio foi utilizar os conceitos de serviços de redes de computadores aliados à configurações de Sistemas Operacionais Linux para mapear uma rede interna utilizando uma solução de monitoramento aberta e distribuída, independente de plataforma, o sistema Zabbix.

Para implementar o Zabbix foi necessário o entendimento da correlação entre os conhecimentos das áreas supracitadas e a estrutura da rede, bem como dos serviços funcionando nesta rede. Este levantamento, junto da necessidade de implantação de um sistema para este tipo de monitoramento, foram de fundamental importância para a escolha do Zabbix, por sua rápida configuração e alta capacidade de personalização e integração a várias plataformas.

Durante a implantação do sistema, houve reuniões de interação entre estagiário, gestores e líderes de equipes de desenvolvimento para que os itens a serem monitorados na rede fossem mensurados quanto à sua importância na estrutura de funcionamento da rede. Este processo contribuiu para uma configuração adequada no que tange a confiabilidade das ações frente ao monitoramento. Deste modo, pôde-se obter um sistema altamente confiável, auxiliando na prevenção e correção de erros, o que garante a alta disponibilidade, fator crucial na tomada de decisões para a implantação do monitoramento.

A empresa tem como objetivo a inovação e qualidade e propiciou ao estagiário a experiência em diversas áreas da Ciência da Computação, pois trata-se de uma empresa com produtos Web. Portanto, acumulou-se experiência desde como os serviços são executados em servidores de aplicação Web, abordando toda a infraestrutura de sistemas e rede até a implementação de um sistema utilizando tecnologias robustas e atuais. Notou-se também uma liberdade para aplicação de novos conhecimentos e comunicação direta com diretores, permitindo a pesquisa e inovação inseridas diretamente no mercado.

Além deste capítulo, este trabalho está dividido em outros 4 capítulos. No Capítulo 2, é apresentada uma visão geral do ambiente de trabalho, incluindo histórico e divisão administrativa da empresa, bem como seus produtos. No Capítulo 3, é apresentada a revisão bibliográfica das áreas relacionadas às atividades executadas no estágio. As atividades desenvolvidas para implantação do serviço de monitoramento são descritas no Capítulo 4, detalhando o levantamento de dados para planejamento da implantação do monitoramento, definições técnicas de criticidade dos itens a serem monitorados, instalação e configuração do Zabbix e suas integrações às plataformas e, por fim, implementação de um proxy para evitar a sobrecarga do monitoramento. Por fim, no Capítulo 5, apresentam-se as considerações finais deste trabalho, incluindo as principais dificuldades encontradas durante as atividades realizadas, a contribuição destas atividades para a formação do estagiário e as conclusões sobre o monitoramento de redes e serviços e a relação as atividades desenvolvidas e a formação em Ciência da Computação.

2 Descrição do Local do Estágio

2.1 Considerações Iniciais

As atividades foram realizadas em Ouro Preto, município localizado na região central do estado de Minas Gerais, onde está sediada a Gerencianet Pagamentos do Brasil. A Gerencianet, como é comumente conhecida, é uma empresa privada de intermediação de pagamentos online, ou seja, entrega no mercado soluções para que um comprador possa pagar um vendedor e este último, por sua vez, seja capaz de gerenciar todas as suas finanças online, de qualquer lugar em qualquer dispositivo ou plataforma, uma vez que o serviço é baseado em aplicações Web. A empresa possui duas sedes, nas quais foram desempenhadas as atividades descritas neste relatório.

Este capítulo apresenta informações sobre a empresa. A seção 2.2 aborda um breve histórico da empresa até chegar nos dias atuais. A seção 2.3 apresenta o espaço físico onde as atividades foram desempenhadas. A seção 2.4 resume a organização administrativa da empresa. A seção 2.5 apresenta os produtos desenvolvidos e entregues no mercado.

2.2 Histórico

A Gerencianet foi fundada em 2007 pelo atual gestor e sócio majoritário Evanil Rosano de Paula e oferece soluções em pagamentos e cobranças e gestão de clientes. Tem sua sede na cidade de Ouro Preto, Minas Gerais e conta, atualmente, com uma carta de clientes e parceiros na ordem dos 30.000. O fluxo anual de transações supera o número de um milhão.

O foco da empresa é descomplicar a intermediação de pagamentos entre vendedor e comprador por meio de soluções amigáveis do ponto de vista de interface e sistema. Ela tem uma visão bem definida e ousada, propiciando aos seus funcionários a motivação necessária para atuar no mercado de trabalho e exigindo dos mesmos a vontade de sempre se superarem em seu cotidiano.

Um dos pontos interessantes do início da empresa está no fato de seu fundador ter formação em Administração pela Universidade Federal de São João del-Rei (UFSJ) e, apesar de a grade deste curso não oferecer habilidades em programação de computadores, estudou e começou a implementação da primeira versão do sistema sozinho, habilidade esta que dificilmente é atribuída a um egresso de curso de Administração.

Atualmente a empresa conta com uma equipe diversificada, incluindo graduados, pós-graduados e mestres em todos os ramos que envolvem desenvolvimento web, possibili-

tando entregar uma solução *e-commerce* completa que atende a todo o Brasil e compete em igualdade com seus concorrentes maiores, no que diz respeito às soluções oferecidas. Devido ao sistema seguro que coloca no mercado, consolidou-se neste segmento antes dominado apenas por grandes empresas multi-nacionais.

2.3 Descrição Física

A Gerencianet situa-se na cidade mineira de Ouro Preto, onde foi fundada em 2007. A cidade, além de universitária, tem sua economia baseada no turismo e na mineração e fica a 90 km da capital Belo Horizonte, o que facilita seu desenvolvimento.

A empresa possui 2 sedes situadas em pontos distintos da cidade. A primeira sede abriga o *Datacenter* próprio da empresa, cuja infraestrutura permite aos clientes o acesso ao sistema e também aos desenvolvedores o acesso aos ambientes de desenvolvimento. A segunda sede é onde ficam todos os funcionários e a infraestrutura de rede e sistemas é acessada remotamente.

Cada desenvolvedor possui uma estação de trabalho que utiliza uma tecnologia de doca, ou seja, cada desenvolvedor possui seu notebook, cedido pela empresa, uma doca, 2 monitores com tela de alta resolução, teclado e mouse. Ao chegar no ambiente de trabalho, basta o desenvolvedor encaixar seu notebook na sua doca e tudo estará funcionando. Se necessário, o desenvolvedor pode levar o notebook para casa, onde pode trabalhar remotamente via conexão autorizada e criptografada, seguindo critérios de segurança internos da empresa.

No caso específico da equipe de Infraestrutura, departamento onde as atividades do presente relatório foram desenvolvidas, a empresa deposita nos membros da equipe a confiança plena para manter o sistema no ar, sendo estes membros responsáveis por todos os ativos que envolvem o provimento dos serviços da empresa disponíveis na Internet.

2.4 Organograma Administrativo

A empresa nasceu como uma *startup*, tendo iniciado seus trabalhos sob demanda de mercado. Assim, a organização administrativa veio com o tempo e o aprendizado. O que se tem hoje é uma tendência de mercado, sendo o fundador da empresa o atual CEO, que tem atuação próxima a todos os setores.

A Figura 2.1 mostra o organograma administrativo dos setores da empresa como um todo. Pode-se notar como os setores de mesmo nível, para interagir entre si, devem comunicar aos setores superiores, mantendo, assim, a constante notificação entre todas as pessoas da empresa.

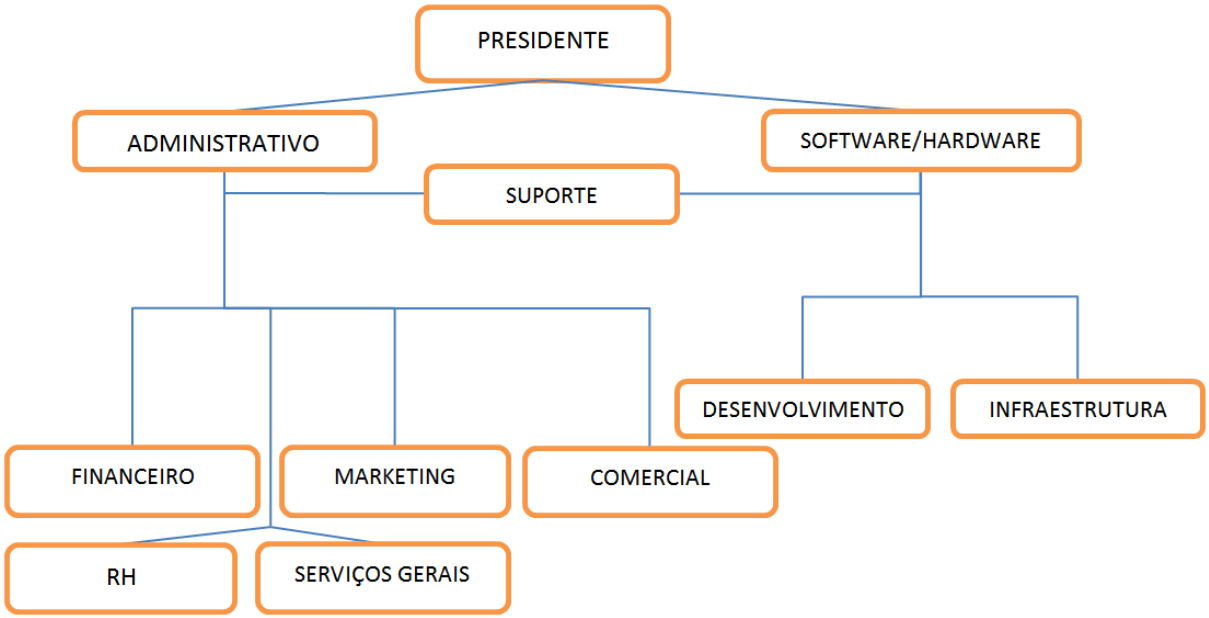

Figura 2.1 – Organograma administrativo da Gerencianet.

2.5 Plataforma de Produtos

A empresa possui atualmente 3 produtos no mercado. São eles:

- *Gerencianet Pagamentos*: Permite a gestão de cobranças e pagamentos virtuais, podendo inclusive enviar cobranças por email, gerar mecanismos de pagamento para colocar em lojas virtuais e *websites*, pagamentos recorrentes, integração com sistemas *opensource* e quaisquer *websites* que queiram vender produtos/serviços. Disponível em: <www.gerencianet.com.br>;
- *Fortunus*: Permite a emissão de boletos e carnês como formas de pagamentos, que são enviados via e-mail ou correios. Emissão em lote, importação de planilha excel para gerar cobranças. Disponível em: <www.fortunus.com.br>;
- *Kuiper*: Permite a criação de lojas virtuais, bem como sua gestão e controle de produtos. Cada loja pode ser integrada ao *Google Analytics* facilitando a melhoria das técnicas sobre como fazer uma página da internet ser encontrada nas primeiras ocorrências nos mecanismos de busca, como *Google*, *Bing*, etc. (SEO), integração com correios, domínio próprio com 3 contas de email. Disponível em: <www.kuiper.com.br>;

Os três produtos da empresa estão disponíveis na internet com tecnologia de desenvolvimento atual e uma infraestrutura própria, mantida pela própria empresa, incluindo

um AS[1] e blocos de endereços IP próprios junto ao <registro.br>, o que permite aos produtos terem seus adicionais, como servidores de resolução de nome (DNS), servidores de email, autonomia em alterações junto à provedores de serviços, entre outros.

2.6 Considerações Finais

Neste capítulo foi apresentada, em diversos âmbitos, a empresa Gerencianet Pagamentos do Brasil, onde as atividades deste trabalho foram desenvolvidas. Foi contada a história da empresa, abordando sua criação, seu nicho de atividade no mercado, o local físico onde as atividades de estágio foram desenvolvidas e os produtos entregues pela empresa no mercado. Estas informações são necessárias para a descrição detalhada das atividades nos capítulos subsequentes.

[1] *Autonomous System*: coleção de prefixos de roteamento conectados por IP sob o controle de um ou mais operadores de rede, sendo que este último define e apresenta uma política de roteamento claramente definida para a Internet (Hawkinson e Bates (1996), Seção 3).

3 Revisão de Literatura

3.1 Considerações Iniciais

O monitoramento de uma rede de computadores, bem como dos sistemas que dependem dessa rede, nasceu com a necessidade de identificação de problemas antes mesmo que eles aconteçam, observando, de maneira automatizada, o comportamento de ativos que desempenhem papéis dentro de uma rede e fazem com que serviços funcionem. Para uma empresa, identificar problemas em estágio inicial e manter a disponibilidade do serviço é crucial para aumentar sua lucratividade (Drilling (2014)).

As atividades desempenhadas e descritas neste relatório se mostram necessárias em qualquer cenário onde se tenha uma rede de computadores. A implementação de sistemas de monitoramento de rede e serviços não é uma tarefa simples, pois nem todo elemento da rede tem uma interface para comunicação com protocolos de gerência de redes (Carneiro, Fortuna e Ricardo (2009)).

Este capítulo visa a apresentar alguns conceitos de redes de computadores, a importância do monitoramento de uma rede, bem como de serviços que executam nesta rede. Estas informações básicas formarão embasamento teórico necessário para a explanação do conceito de monitoramento de rede e sistemas. A seção 3.2 apresenta breves conceitos de Gerência de Redes de Computadores com embasamento principal no livro Kurose e Ross (2010). A seção 3.3 aborda os princípios dos sistemas de monitoramento de redes, bem como arquiteturas para que os mesmos funcionem.

3.2 Gerência de Redes

De acordo com Mendes (2007), rede de computadores é um termo que se atribui a um conjunto de processadores capazes de trocar informações e partilhar recursos, interligados entre si de modo que compartilham recursos físicos e lógicos, sendo estes de vários tipos: dados, impressoras, mensagens, etc. Este conjunto de processadores possui equipamentos que são comumente categorizados como ativos ou passivos, *i.e.*, equipamentos que podem ser configurados (Exemplos: roteadores, computadores, servidores, dentre outros) e equipamentos não configuráveis (*hubs*, *switches*, cabos de rede, dentre outros), respectivamente (Amaral e Dias (2011)). Disto, nasceu a internet, que hoje provavelmente é o maior sistema de engenharia já criado pelo homem, com centenas de milhões de componentes interligados, incluindo os milhões de usuários que fazem uso desta grande rede (Kurose e Ross (2010)).

Em 27 de outubro de 1980, os pesquisadores da ARPAnet[1] sentiram pela primeira vez a necessidade de um sistema de gestão da sua grande rede (Rosen (1981)). Por várias horas, a rede que funcionara há anos sem nenhum problema sofreu uma queda, devido a um processo de software rodando sem controle. Por causa desta queda, muitas pessoas envolvidas na ARPAnet se interessaram em entender o problema e cuidar para que o mesmo não acontecesse novamente. Este acontecimento foi relatado na RFC789 (Rosen (1981)), como forma de ajudar nas pesquisas de um sistema que auxilie a prever tais sinistros, bem como corrigir as falhas dos algoritmos que fizeram com que a rede parasse de funcionar.

Segundo Saydam e Magedanz (1996), "Gerência de rede inclui o oferecimento, a integração e a coordenação de elementos de hardware, software e humanos, para monitorar, testar, consultar, configurar, analisar, avaliar e controlar os recursos da rede, e de elementos, para satisfazer às exigências operacionais, de desempenho e de qualidade de serviço em tempo real a um custo razoável". Kurose e Ross (2010) citam alguns tópicos de fundamental interesse para a Gerência de redes. São eles:

- Identificação e gerenciamento de falhas;
- Detecção proativa de anomalias;
- Correlação entre alarmes;
- Gerenciamento de serviços.

Na Figura 3.1 é apresentada uma arquitetura básica para um sistema de gerência de redes, onde há três componentes principais: uma entidade gerenciadora, os dispositivos gerenciados e um protocolo de gerenciamento de rede. De fato, como uma rede é composta por ativos e passivos de diversos tipos, esta é tida como distribuída. Sendo assim, é necessária a coleta de dados por uma entidade central que gerencia tais dados e fornece informações a um administrador de rede.

[1] ARPAnet é uma das primeiras redes de computadores do mundo, criada pela ARPA (*Advanced Research Projects Agency*, ou Agência de Projetos de Pesquisa Avançada, em tradução livre, uma agência do Departamento de Defesa dos EUA). Nela foram testados conceitos de redes inovadores, como: comutação de pacotes, topologia e roteamento distribuídos, e interligação entre vários sistemas de computadores (Abbate (1994)).

Figura 3.1 – Componentes de uma arquitetura de Gerência de redes (adaptado de Kurose e Ross (2010))

Entidade Gerenciadora é uma aplicação que deve "enxergar"toda a rede automaticamente e tem um ser humano para seu controle. Ela controla a coleta, o processamento, a análise e a apresentação de informações de gerenciamento da rede, utilizando um protocolo de gerenciamento de rede. A partir destas informações, provenientes dos dispositivos gerenciados, são tomadas as decisões automatizadas ou não sobre o comportamento da rede.

Dispositivo gerenciado é um ativo de rede (incluindo seu software) ou um servidor que possuem em si um agente instalado. Este agente coleta dados de funcionamento do dispositivo e os prepara para serem processados pela entidade gerenciadora, por meio do protocolo de gerenciamento de rede. É neste agente que pode-se criar formas de gerenciamento de situações que são específicas de cada dispositivo, *e.g.*, monitorar uma tabela de um banco de dados MySQL rodando em um servidor Linux.

Protocolo de gerenciamento de rede é o que permite a "conversa"entre a entidade gerenciadora e o dispositivo gerenciado. Este protocolo permite que o administrador da rede configure o dispositivo gerenciado para enviar informações à entidade gerenciadora e fornece uma ferramenta para monitorar, testar, consultar, configurar, analisar, avaliar e controlar a rede.

No final da década de 1980, os pesquisadores amadureceram os padrões para gerência de rede, com o OSI **CMISE/CMIP** (*Common Management Service Element/Common Management Information Protocol*) (Piscitello e Chapin (1993), Stallings (1993), Glitho (1998)), tido como o protocolo de acesso mais poderoso existente até uma certa época

(Ellanti (2005)) e o **SNMP** (*Simple Network Management Protocol*, ou Protocolo Simples de Gerenciamento de Rede) (Case et al. (2002), Stallings (1998), Rose (1996)), que, por ter sido projetado e oferecido quando a necessidade de gerência de redes crescia, passou a ser o mais utilizado (Lopes e Oliveira (1997)). Estes padrões foram projetados para funcionarem de maneira genérica, em elementos de rede de qualquer fabricante, porém o CMISE/CMIP ficou sendo mais utilizado na área de telecomunicações e o SNMP tornou-se popular para redes de computadores locais (Junior e Rochol (1998)).

O gerenciamento de redes, atualmente, é uma evolução dos itens abaixo, em ordem evolutiva, com foco cada vez mais nas necessidades dos clientes, buscando excelência operacional:

- Gerenciamento de falhas;
- Gerenciamento de desempenho;
- Gerenciamento de serviços;
- Gerenciamento de redes.

Para que o administrador de rede possa garantir estabilidade (todos os elementos em estado de funcionamento normal), alta disponibilidade, segurança e eficiência da rede, torna-se essencial um ambiente de gerenciamento flexível, que possa ser rápida e facilmente adaptado para monitorar cenários cada vez mais dinâmicos (Gaspary et al. (2001)). Para isso, é fundamental a escolha adequada de uma ferramenta que atue nos diversos ítens que abrangem a gerência de redes.

3.3 Sistemas de monitoramento de redes

Para entender o comportamento e garantir qualidade de serviço de uma rede, é essencial a busca pela acurácia na medição desta rede (Caceres et al. (2000)). Com o aumento na largura de banda e a entrega de redes com maiores capacidades em qualidade de serviço, as pesquisas em monitoramento de rede tornaram-se críticas (Lai e Baker (1999)).

Diversas ferramentas de monitoramento estão disponíveis atualmente, sendo elas livres ou pagas. Algumas empresas contratam outras empresas – estas últimas especializadas em monitoramento – e desenvolvem suas próprias ferramentas ou utilizam um conjunto de ferramentas existentes para tal serviço. Os sistemas disponíveis facilitam a vida do administrador de sistemas e redes por meio de métricas e informações que auxiliam a correção de problemas e tomadas de decisões relacionadas ao futuro da rede que ele gerencia (Dias (2008)).

Visando a organizar e escolher o melhor sistema de monitoramento, algumas características devem ser levantadas para tal tomada de decisão. Seguem os critérios estipulados pelos administradores da rede da empresa:

- Facilidade de utilização da ferramenta;
- Facilidade de configuração da ferramenta em um servidor;
- Facilidade da intercomunicação da ferramenta com hardware, software e serviços;
- Capacidade de integração com outras ferramentas auxiliares.

Os sistemas de monitoramento mantêm um rastro do ciclo de vida de um problema, facilitando a resolução do mesmo e também mantêm outros aspectos da rede, tais como modificações de configuração, de segurança, entre outros (Melchiors (1999)). Por este motivo, estes sistemas ou entidades gerenciadoras, são tidos como memória da operação da rede, por terem um histórico de dados coletados que "contam uma história"da rede desde seu nascimento, ou desde a implantação do sistema de monitoramento na rede.

O registro de eventos de um sistema de monitoramento é baseado em um protocolo, geralmente o SNMP, mas apenas dados coletados em um banco de dados significam pouco atualmente. É necessário um sistema de tomada de decisões. Esse é um processo freqüentemente decisivo para determinar o sucesso ou a falha de um ambiente (Melchiors (1999)).

Exemplos de processos de tomada de decisão dentro da gerência de redes incluem gerenciamento da detecção e correlação de falhas na rede, do roteamento/engenharia de tráfego, do planejamento de configuração e controle de configuração online, da análise e otimização do desempenho dos sistemas de comunicação, da análise da segurança da rede, da detecção de intrusão na rede (Ericson, Ericson e Minoli (1989)).

Para a escolha da ferramenta que foi utilizada, três das disponíveis e mais importantes no mercado foram testadas. A seguir, apresenta-se breve descrição destas ferramentas de monitoramento de redes: Cacti, Nagios e Zabbix.

3.3.1 Cacti

Tem como principal função o monitoramento do estado de elementos de rede e programas, analisando também CPU e largura de banda. Porém, após a instalação, a interface não se mostrou amigável e, apesar dos gráficos precisos, são necessários muitas etapas para se chegar a um resultado satisfatório de extração de dados. Na Figura 3.2 tem-se a tela administrativa principal do Cacti.

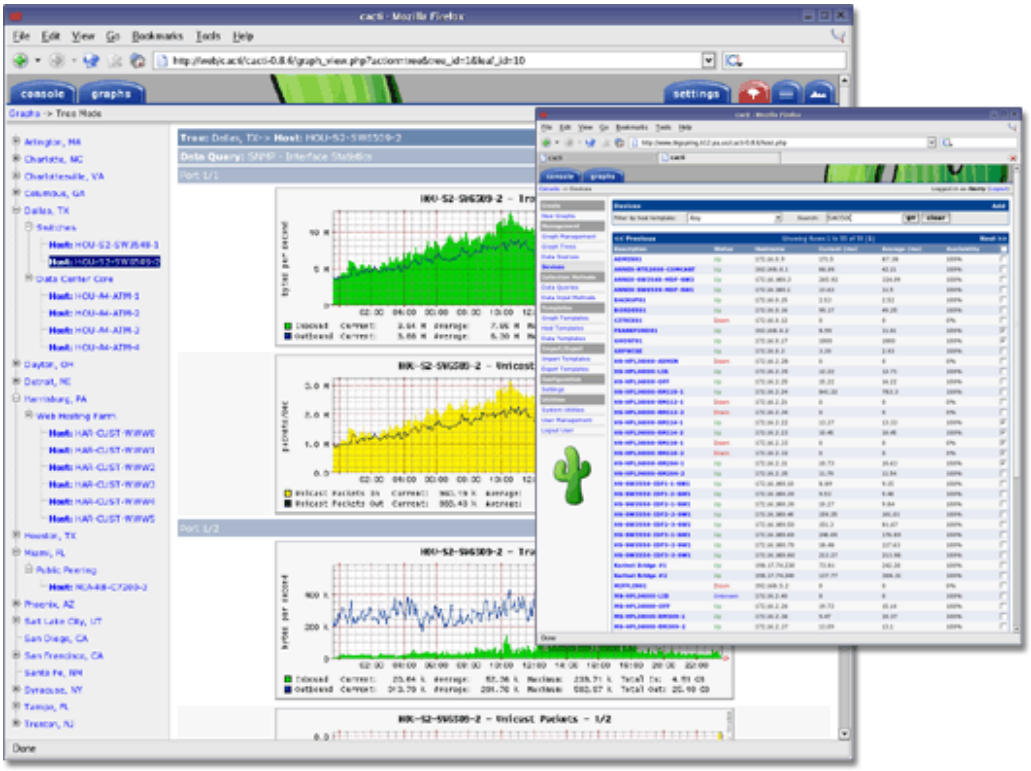

Figura 3.2 – Tela principal do Cacti (Fonte: <http://www.cacti.net/>)

3.3.2 Nagios Core

É o sistema mais popular para classes empresariais, pois integra-se quase que com qualquer hardware/software e serviço, utilizando diversos protocolos de rede. Seu problema principal, após instalação, foi a dificuldade da configuração de plug-ins e aprendizado demorado, o que dificultou sua implementação para soluções rápidas. A tela do Nagios Core pode ser vista na Figura 3.3.

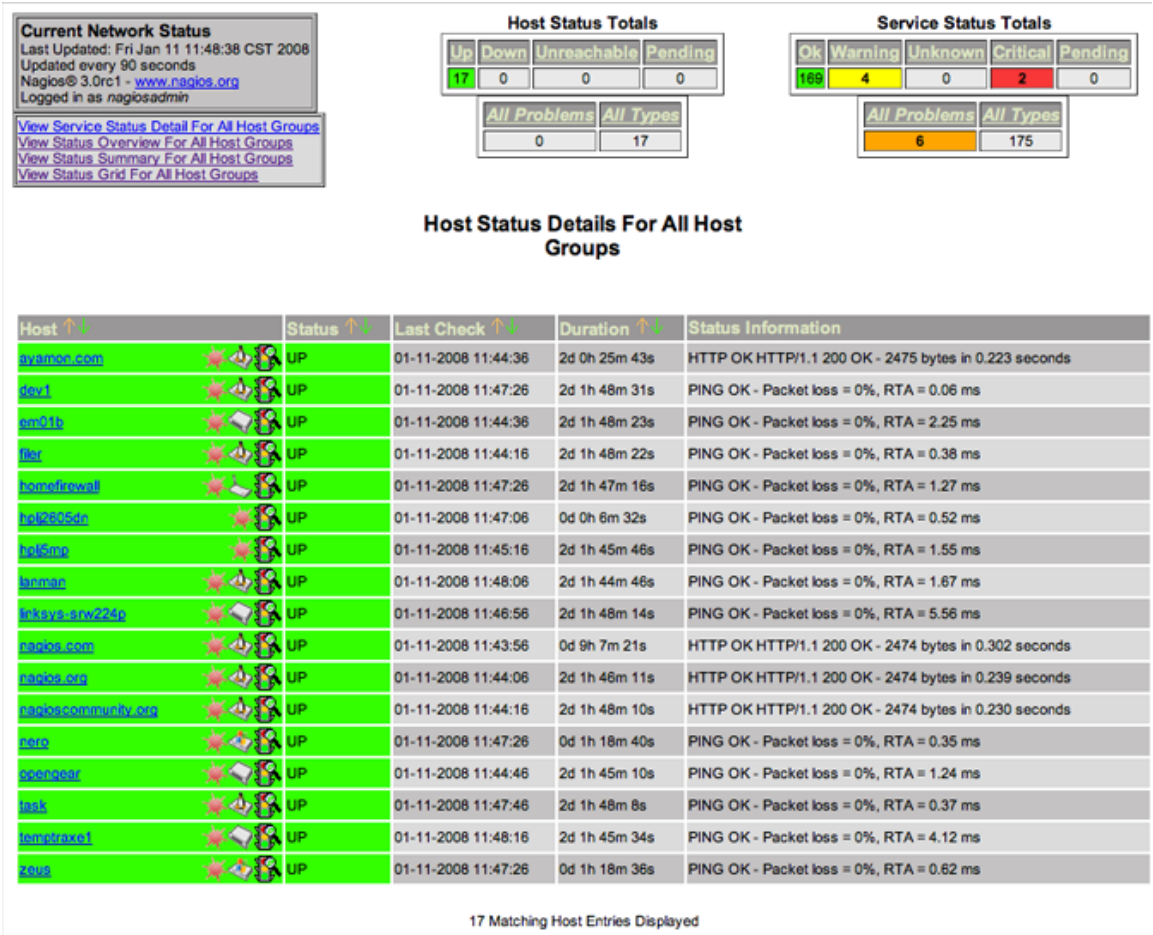

Figura 3.3 – Tela de detalhes de equipamentos monitorados do Nagios Core (Fonte: <http://www.nagios.com/products/nagioscore/screenshots>)

3.3.3 Zabbix

É um sistema de monitoramento amigável que permite uma fácil configuração. Para o monitoramento de serviços, é necessária a instalação de um agente no servidor onde deseja-se realizar o monitoramento e, com este agente, utiliza-se *templates* para monitoramento do estado do servidor, como CPU, memória e disco, além de ser possível implementar *scripts* para serviços mais específicos.

Além disso, ele é baseado em *triggers*, ou seja, é possível a configuração de monitoramento de um tipo de serviço ou ativo de rede, com isso cria-se um estado para que o Zabbix tome providências a respeito deste estado. As *triggers* facilitam as notificações quando algum serviço sai de seu funcionamento padrão, podendo emitir alertas via e-mail, *chat* e até SMS (mensagens de texto). A tela administrativa do Zabbix é mostrada na Figura 3.4.

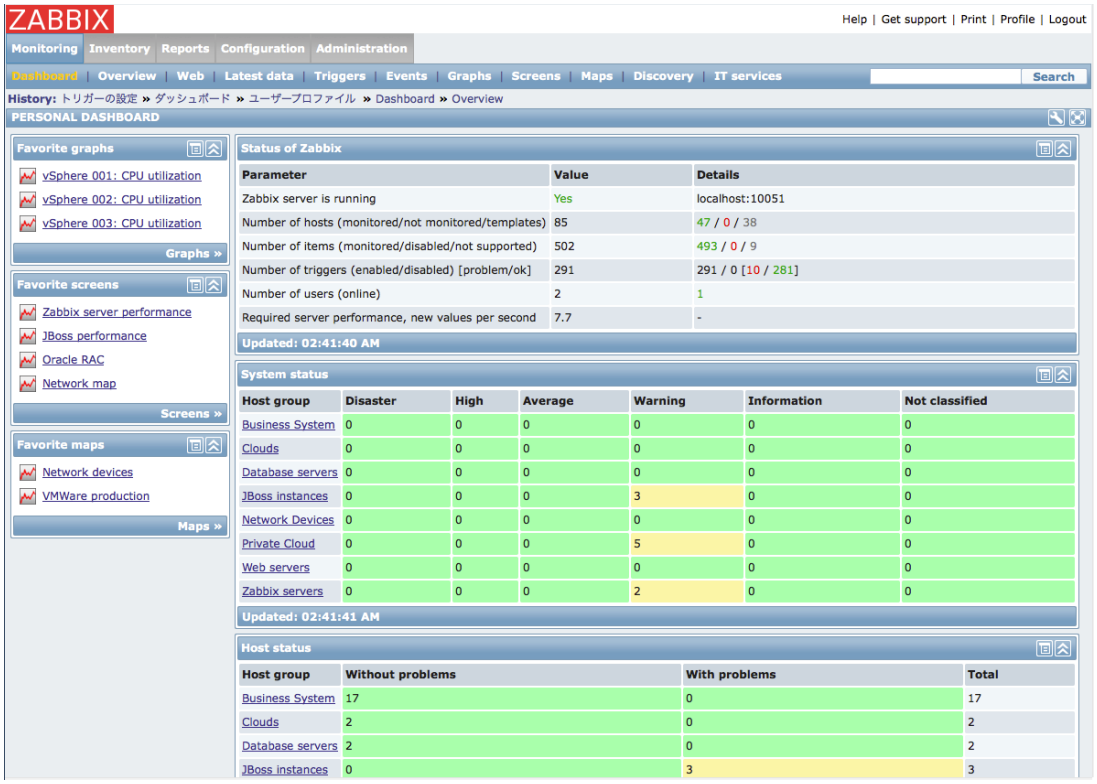

Figura 3.4 – Tela administrativa principal do Zabbix (Fonte: <http://www.zabbix.com/screenshots.php>)

3.4 Considerações Finais

Neste capítulo foi apresentada a fundamentação teórica de Gerência de Redes baseada, principalmente, no livro Kurose e Ross (2010), abordando a generalidade de protocolos que surgiram no final da década de 1980 e permanecem até os dias atuais, permitindo a interoperabilidade entre equipamentos, independente de seus fabricantes e/ou sistemas. Também foram discutidos os princípios de funcionamento de sistemas de monitoramento de forma geral, pois estes são implementados sobre os mesmos protocolos definidos anteriormente.

Tais conceitos introduzem a importância de ferramentas que auxiliem o administrador de redes e de sistemas a prever sinistros em ativos de rede ou servidores que estejam em funcionamento em sua rede. Além do mais, estes cuidados ajudam as empresas a realizarem investimentos para que prejuízos maiores sejam evitados.

4 Atividades desenvolvidas

4.1 Considerações Iniciais

Ao longo do período de estágio, as atividades previstas foram realizadas, supervisionadas, orientadas e catalogadas a fim de manter-se um acompanhamento entre os membros envolvidos e também cumprir legislação de estágio vigente. Este capítulo tem como objetivo apresentar tais atividades.

A seguir, serão apresentadas na seção 4.2 as atividades realizadas com o total de horas em cada atividade e, na seção 4.3 a descrição detalhada de cada atividade.

4.2 Cronograma de atividades

As atividades aqui descritas foram propostas no início do estágio, quando apresentou-se um cronograma de horas para cada atividade. Abaixo, são descritas as atividades desenvolvidas e entre parênteses estão as referências para as subseções desta seção que descrevem cada atividade:

1. Levantamento de hardware, software, servidores e serviços presentes na rede (4.3.1)
2. Definição da criticidade dos serviços monitorados (4.3.2)
3. Especificação das características intrínsecas a cada sistema que foi monitorado (4.3.3)
4. Implementação e configuração do servidor Zabbix (4.3.4)
5. Levantamento e configuração de ações tomadas (4.3.5)
6. Configuração de *triggers* para cada tipo de monitoramento realizado (4.3.6)
7. Configuração de integração com sistemas paralelos (4.3.7)
8. Verificação da solidez do monitoramento (4.3.8)
9. Implementação de servidor Zabbix proxy para diminuir carga de processamento (4.3.9)

A Tabela 4.1 mostra a relação entre atividades (linhas) e semanas em que foram realizadas estas atividades (colunas).

Atividade/Semana	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1										
2										
3										
4										
5										
6										
7										
8										
9										

Tabela 4.1 – Cronograma de atividades

Para efeito de explicação, as semanas enumeradas nas colunas da Tabela 4.1 são explicitadas abaixo:

1. 15/set - 19/set
2. 22/set - 26/set
3. 29/set - 03/out
4. 06/out - 10/out
5. 13/out - 17/out
6. 20/out - 24/out
7. 27/out - 31/out
8. 03/nov - 07/nov
9. 10/nov - 14/nov
10. 17/nov - 21/nov

4.3 Descrição das atividades desenvolvidas

4.3.1 Levantamento dos elementos a serem monitorados

Para monitoramento da rede, é necessário saber o que se pode monitorar, o que se deve monitorar e o que é imprescindível o monitoramento. Este levantamento é crucial para que exista alta disponibilidade dos serviços fornecidos por esta rede, bem como identificação prévia e correção de falhas. Na Figura 4.1 está representada a estrutura da rede que será monitorada e cada ativo de rede presente na figura é descrito a seguir.

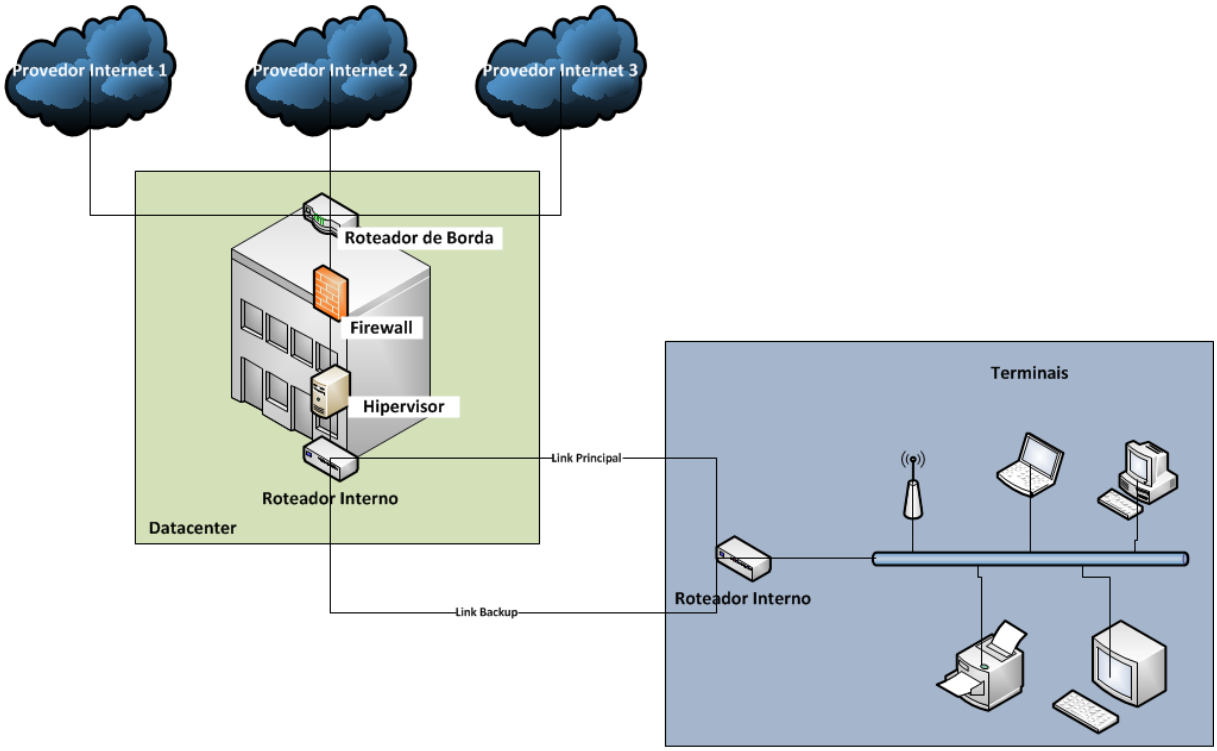

Figura 4.1 – Estrutura básica da Rede monitorada.

- Roteador de borda: Roteador com sistema operacional e hardware avançados para conexão com diferentes provedores de internet.
- Firewall: Servidor que gerencia a saída e entrada de pacotes para a rede interna e entre roteadores internos.
- Hipervisor: Servidor que roda um sistema operacional de virtualização, ou seja, é uma camada de software hospedeira de máquinas virtuais (Servidores Virtualizados) que possuem o mesmo conjunto de instruções onde está sendo executado este hipervisor (Bressoud e Schneider (1995)).
- Roteador interno: Roteadores para gerência interna de pacotes e políticas de acesso.
- Terminais: Sistemas finais utilizados pelos profissionais da empresa, como estações de trabalho, notebooks, impressoras, acessos sem fio, *smartphones*, televisores, etc.

Neste cenário, separou-se, então, todos os ativos da rede que precisariam ser monitorados e a prioridade dos itens de cada ativo que seria monitorado. A Tabela 4.2 mostra a categorização final tida como essencial para manter o acesso externo de clientes da empresa e também os funcionários em pleno conforto de utilização. Esta categorização teve como princípio a alta disponibilidade da rede e dos serviços que nela são disponibilizados.

Ativo de rede	**O quê monitorar**
Nobreaks	Origem da energia (rede elétrica ou bateria)
Roteador de borda	Estado das conexões com os provedores de internet, Processador, Memória, Tráfego
Sensor de Temperatura do Datacenter	Temperatura
Servidores Hipervisores	Processador, Memória, Disco, Disponibilidade na rede
Servidores Virtualizados	Processador, Memória, Disco, Disponibilidade na rede, DNS, Web, Banco de Dados, Replicação de Banco de Dados, Alteração de usuários, scripts e serviços específicos

Tabela 4.2 – Descrição dos ativos de rede e os itens a serem monitorados

4.3.2 Definição da criticidade de serviços

Após o levantamento dos ativos da rede e dos itens presentes em cada ativo, foi necessário também estipular a criticidade do cada item relacionado ao ativo em que está presente, ou seja, o quão grave uma variação no item pode ser para o ativo. Para isso, utilizou-se a metodologia do questionamento. Por exemplo: "o item **CPU** do ativo de rede **Roteador de borda**, se sofrer muita variação, afetará o funcionamento do ativo e também dos dependentes do ativo?".

O julgamento da criticidade, ou seja, se é crítico ou não crítico, foi baseado, mais uma vez, na alta disponibilidade do sistema como um todo. Foram definidos os níveis de criticidade: baixa (B), média (M) e alta (A). Estes níveis podem ser expandidos juntamente com o crescimento da empresa e da rede, para que se granularize a ação em caso de tomada de decisões ou correção falhas. Para melhor organização da informação numa tabela, enumerou-se os itens da seguinte forma:

1. Origem da Energia (rede elétrica ou bateria)
2. Estado da conexão com os provedores de internet
3. Processador
4. Memória
5. Tráfego
6. Temperatura
7. Disco
8. Disponibilidade na rede
9. DNS

10. Web
11. Banco de Dados
12. Replicação de Banco de Dados
13. Alteração de usuários
14. Scripts
15. Serviços Específicos

	Itens														
Ativo de rede	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Nobreaks	A														
Roteador de borda		A	M	B	A										
Sensor de Temperatura do Datacenter						A									
Servidores Hipervisores			M	M			A	A							
Servidores Virtualizados			M	M			A	A	M	A	A	A	V	V	V

Tabela 4.3 – Nível de criticidade de cada item em seu ativo, onde: A=alta; M=média; B=baixa; V=variável

Na Tabela 4.3 estão descritos os níveis de criticidade estipulados para cada item de cada ativo de rede. Além dos 3 níveis mencionados previamente no texto, nesta tabela está presente o nível V, que significa variável. Este nível está somente representado nesta tabela a fim de sumarização, ou seja, são muitos servidores virtualizados, cada servidor tem um tipo de serviço rodando, cada serviço tem suas especificidades e , como serão descritas posteriormente, em atividades seguintes, nesta tabela não serão aprofundados.

4.3.3 Especificação de atributos dos sistemas a serem monitorados

Nesta atividade, houve reuniões com os líderes das equipes responsáveis por cada produto da empresa, além de sistemas internos e gestores, para definições de tarefas específicas de cada produto/sistema que devem ser monitoradas a fim de prestar um *feedback* interno ou até para o próprio cliente da empresa. Foi realizado um estudo sobre a viabilidade de se monitorar cada tarefa definida.

Nesta fase, estes líderes e pessoas envolvidas ainda estavam em dúvidas quanto às consequências da implantação do monitoramento, uma vez que esta mudança envolve investimento e o retorno deste investimento era, até então, desconhecido. De maneira geral, dentre os monitoramentos, estão os seguintes (por motivos de segurança interna da empresa, não foi possível listar neste documento, de fato, todos os ítens monitorados):

- Integrações entre a empresa e outras empresas
- Tráfegos de internet específicos, com geração de gráficos
- Tráfegos entre servidores críticos
- Sobrecarga de servidores
- Estado da alta disponibilidade da plataforma de produtos
- Saldos de serviços tercerizados utilizados pela empresa

Após algumas reuniões, foi apresentado um relatório (omitido neste documento por causas de segurança) embasado em estudos teóricos em gerência de redes, experiências de implantação em outras empresas e como cada equipamento seria monitorado. Com este relatório, pôde-se demonstrar o poder do monitoramento e selar a confiabilidade neste investimento a longo prazo.

4.3.4 Implementação do servidor Zabbix

O servidor do Zabbix, por motivos de isolamento, segurança e confiabilidade, não foi virtualizado no Datacenter. Utilizou-se para tal um servidor básico, que se encontra, na topologia, juntamente aos terminais. Esta localização é estratégica e garante que qualquer falha nos itens de cada ativo de rede que serão monitorados não afete o monitoramento em si e colete dados chamados de **falso positivos**.

Como o Zabbix é *Open Source*, é possível baixar, além de pacotes pré-compilados, seu código-fonte. Por motivos de aprendizado e estudo para manutenção futura, preferiu-se a instalação compilada. A instalação foi concluída seguindo o manual de instalação contido na documentação *online* do Zabbix.

Foi utilizada a configuração da compilação demonstrada no Comando 4.1. Algumas *flags*[1] presentes requerem a instalação de pacotes antes da execução deste comando. No *CentOS* (instalação mínima[2]), sistema operacional escolhido para a instalação do Zabbix, os principais pacotes necessários (dependendo do sistema, os principais podem depender de pacotes adicionais) seguem:

- gcc (*GNU Compiler Collection*): pacote necessário para compilação de códigos escritos na linguagem C (também compila C++, Objective-C, Fortran, Java, Ada e Go);

[1] *Flags* são argumentos do *script* de configuração necessários para alterar as configurações padrão e habilitar funcionalidades

[2] O *CentOS* oferece a opção de instalação mínima, somente com recursos básicos, como Kernel, vídeo, rede, entrada/saída.

- mysql-server: pacote necessário para executar um servidor MySQL;
- mysql-client: pacote necessário para acessar o servidor MySQL;
- libxml2: pacote necessário para análise da linguagem de marcação XML (utilizada em templates do Zabbix, por exemplo);
- libcurl: pacote necessário para transferências de arquivo e requisições, que suporta diversos protocolos (dentre eles HTTP e HTTPS, utilizados pelo Zabbix);
- net-snmp: pacote necessário para a comunicação entre Zabbix e equipamentos que utilizem o protocolo SNMP.

```
1 # ./configure --enable-server --enable-agent --with-mysql --with-net-snmp
    --with-libcurl --with-libxml2
```

Comando 4.1 – Configuração para compilação do Zabbix Server

O Zabbix possui um serviço chamado *zabbix server* que fica em execução permanente na porta 10051 (padrão), e o agente que se comunica com o servidor (chamado de *zabbix agent*) roda na porta 10050. Ajustadas as configurações de nome do servidor, IP e liberação desta porta no Firewall interno, o serviço pode ser iniciado. Para o gerenciamento deste serviço e dos monitoramentos, existe a interface Web do Zabbix. Como uma das *flags* de configuração incluíam a instalação do agente do Zabbix no mesmo sistema que o servidor do Zabbix, este pode ser o primeiro teste para verificar a instalação do serviço.

Para verificar se correu tudo como planejado, pode-se rodar o Comando 4.3 e o Comando 4.2 no *shell*[3] do Linux onde o Zabbix foi instalado. Se nenhum dos comandos produzir resultados, algo está errado. Neste caso, pode ter faltado algum pacote ou ocorrido algum erro de compilação. Isto deve ser analisado nos logs da compilação.

```
1 # ps axu | grep -i "[z]abbix"
```

Comando 4.2 – Este comando deve gerar uma lista de serviços do zabbix sendo executados

```
1 # netstat -natup | grep -i "[z]abbix"
```

Comando 4.3 – Este comando deve mostrar as conexões de rede que estão sendo utilizadas pelos serviços do servidor Zabbix

Para que o *zabbix server* se comunique com um ativo de rede a ser monitorado, é necessário utilizar algum protocolo padrão ou instalar um agente (*zabbix agent*) neste ativo. Na grande maioria dos casos, pode-se instalar o agente do Zabbix, o que permite a escolha de quais ítens devem ser monitorados no ativo. O Zabbix trabalha com dois tipos de verificação, descritas a seguir:

[3] *Shell* é um conjunto de instruções (programa) que oferece uma interface baseada em texto para Linux e outros sistemas baseados em Unix. Tem como função principal a leitura de comandos digitados dentro de um console (*i.e.*, um modo somente texto) em uma interface gráfica, e então executá-los (LINFO (2006)).

1. Passiva: Neste tipo de verificação, o agente responde a uma requisição do servidor, ou seja, o servidor (ou *proxy*, se houver) requisita uma informação (porcentagem de uso do processador, por exemplo) e o agente envia o resultado como resposta

2. Ativa: Neste tipo de verificação, o agente recebe do servidor uma lista de itens a terem seus dados coletados, processa esta informação e as envia. A partir daí, enviará periodicamente novos valores para o servidor

Pelo fato de ser mais simples, escolheu-se a verificação passiva. Assim, foi instalado um agente do Zabbix em cada ativo da rede, onde era possível ser instalado. Para instalar o agente, seguiu-se também a documentação oficial, cuja configuração é semelhante à instalação do servidor e pode ser vista no Comando 4.4.

```
1 # ./configure --enable-agent
```

Comando 4.4 – Configuração para compilação do Zabbix Agent

4.3.5 Levantamento de ações para monitoramento dos elementos

Com o Zabbix em execução em um servidor isolado da infraestrutura central, *i.e.*, alocado fora do Datacenter, nesta atividade, foi necessário um planejamento de ações a serem tomadas a partir dos dados coletados pelo Zabbix. Cada usuário no Zabbix deve possuir mídias configuradas. Estas mídias, como definido no manual oficial do Zabbix, são métodos de entrega de notificação, por exemplo: email, mensagens de texto (SMS) e mensagem instantânea.

Esta configuração foi feita na interface de gerenciamento do Zabbix, ilustrado na Figura 4.2. Na configuração da mídia, já deve ser definido qual o tipo de severidade deve ser notificada a este usuário por meio desta mídia, bem como a disponibilidade da notificação.

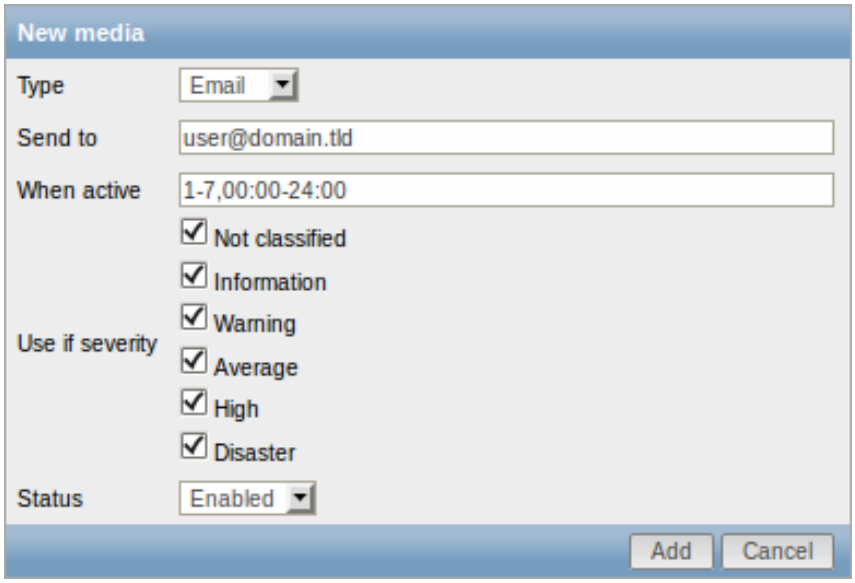

Figura 4.2 – Adicionando um tipo de mídia (Email) para um usuário exemplo

4.3.6 Configuração de *triggers*

Como já citado anteriormente, o Zabbix coleta dados. Este recebimento e armazenamento de dados funciona apenas como registro de atividades de um item (porcentagem de uso da memória, por exemplo), se nenhuma análise for feita em cima destes dados coletados. Após um item ser criado, como na Figura 4.3, este deve ser configurado para ser avaliado e emitir alertas a partir de limiares, ou seja, valores mínimo, máximo, padrões, etc.

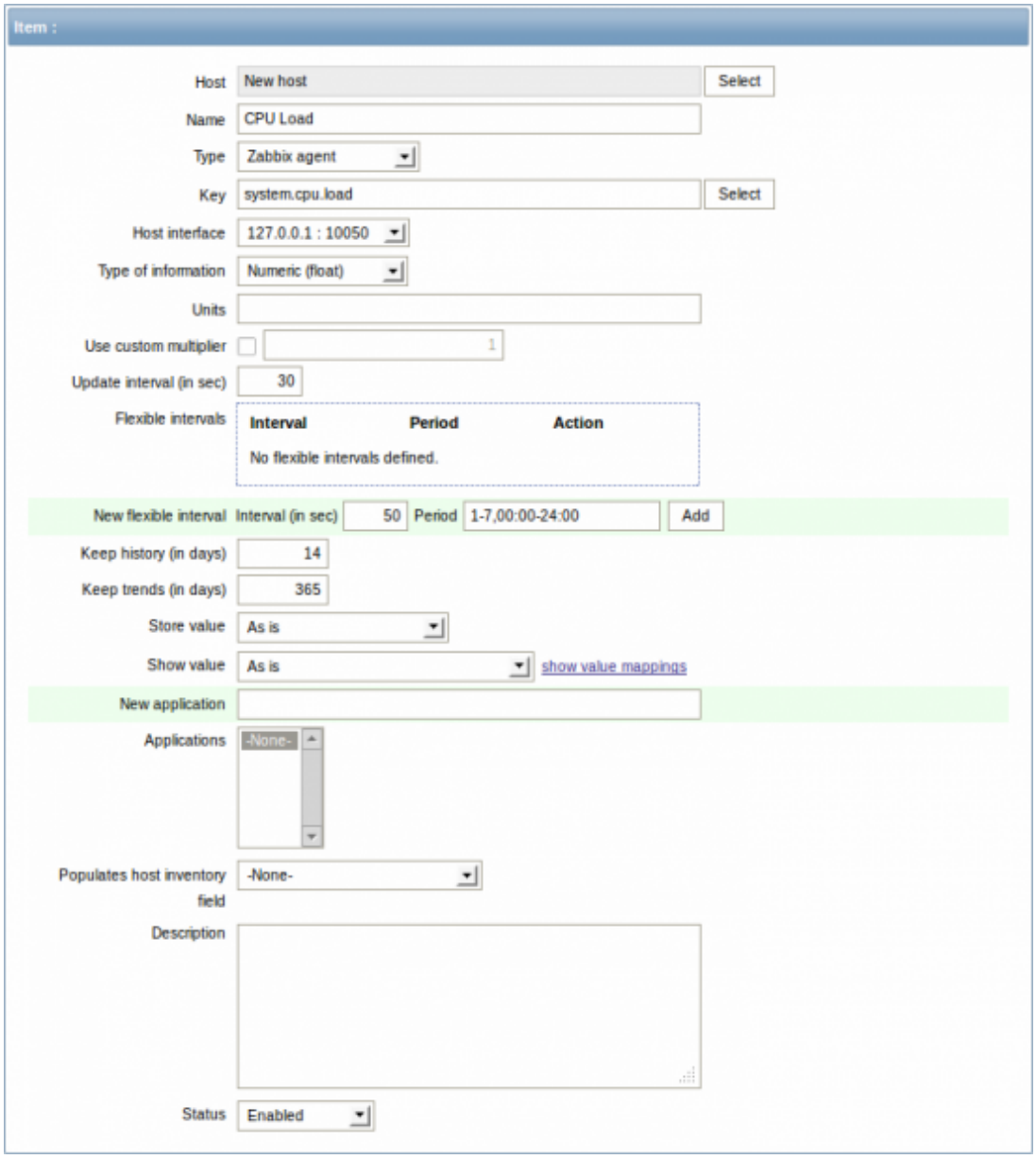

Figura 4.3 – Adicionando um item (Carga de processamento) para o ativo de rede 127.0.0.1 (local)

A partir do momento em que um item é configurado, o Zabbix Server tenta conexão com o Zabbix Agent requisitando valores para aquele item. Estes valores são coletados periodicamente, de acordo com intervalo configurado na adição do item. Para definir um *threshold*[4] do que é aceitável ou não, adiciona-se uma *trigger* para o item do ativo de rede.

A *trigger* é baseada em uma expressão lógica a respeito de valores monitorados. A expressão é do tipo:

{<servidor>:<item>.<função>(<parâmetro>)}<operador><constante>

Onde:

[4] *Threshold*, em tradução livre, significa limiar, ou seja, o ponto que separa uma classificação de outra.

- servidor: Nome ou IP do ativo de rede que foi cadastrado no Zabbix Server
- item: Item adicionado no ativo de rede
- função: Função (soma, média, subtração, contador, etc) suportada pelo Zabbix [5]
- parâmetro: Argumento aceito pela função
- operador: Operadores lógicos e aritméticos
- constante: número (inteiro, decimal, etc) ou *string*

A Figura 4.4 mostra a tela de adição de *trigger* para um item do ativo de rede. Nela pode-se ver o campo de expressão, que pode ser bastante longa e, muito importante, a classificação da severidade, comentada em atividade anterior. Para que a modularização da implantação do Zabbix perpetue, *i.e.*, a configuração feita inicialmente manter-se organizada, esta classificação deve ser feita de maneira correta a fim de garantir a confiabilidade do projeto. Se um serviço está sendo testado, mas configura-se uma *trigger* com severidade "Desastre"e o Zabbix emite um alerta para algum usuário, este pode deslocar-se para resolver um problema que não existe. Este deslocamento, para a empresa, é um prejuízo.

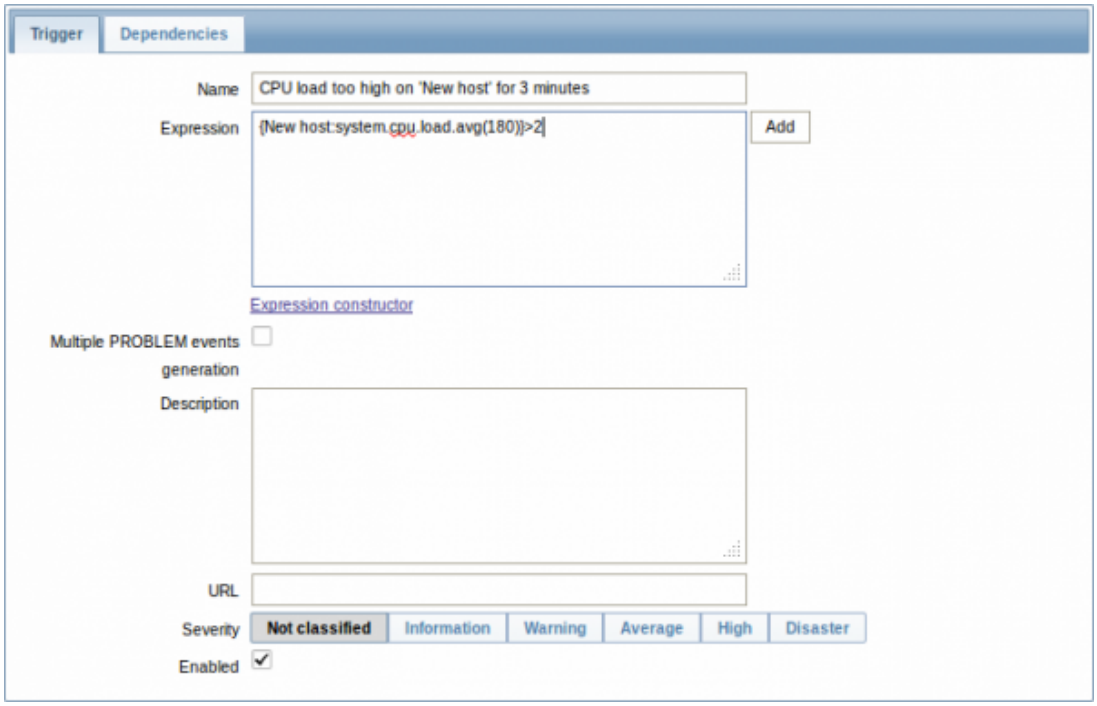

Figura 4.4 – Adicionando uma *trigger* que verifica se a carga de processamento nos últimos 3 minutos (180 segundos) é maior que 2

Sendo estes termos esclarecidos, o que foi feito na rede da empresa foi basear-se em *templates*, que são agrupamentos de itens e *triggers* que podem ser utilizados para

[5] Funções aceitas pelo Zabbix 2.4: <https://www.zabbix.com/documentation/2.4/manual/appendix/triggers/functions>

vários ativos de rede, não sendo necessário configurar o mesmo item e a mesma *trigger* várias vezes. O Zabbix, por padrão, possui alguns *templates* que podem ser baixados e personalizados de acordo com a demanda da rede ou do serviço.

A princípio, foram configurados no Zabbix todos os ativos de rede da empresa cujo sistema operacional é Linux e associou-se a eles o *template* padrão para Sistemas Operacionais Linux. Porém, o arquivo de configuração do Zabbix Agent, situado em cada ativo de rede, permite que seja configurado um comando qualquer que retorne um valor diferente de padrões.

Em suposição, existe um ativo de rede (Servidor Linux Teste) que executa um serviço chamado 'teste'. Deseja-se configurar no Zabbix Server um item e uma *trigger* para saber qual a porcentagem de consumo de processamento do serviço 'teste'. Pode-se adicionar no final do arquivo de configuração do Zabbix Agent que roda naquele ativo a linha abaixo:

```
1 UserParameter=system.servico.teste[*], (ps axu | grep "[t]este" | awk '{
    print $3}')
```

Comando 4.5 – Parâmetro personalizado no arquivo de configuração do Zabbix Agent

Desta forma, criaria-se uma *trigger* cujo nome seria, por exemplo "Serviço teste está sobrecarregando Servidor"e a expressão seria como abaixo:

{{Servidor Linux Teste:system.servico.teste.last(0)}>10}

4.3.7 Integração com sistemas paralelos

Esta atividade relata a necessidade da integração entre o Zabbix Server e plataformas que auxiliam na performance de seus recursos. Para elucidar, tem-se os exemplos dos alertas, que dependem de alguma ferramenta externa para funcionarem com confiabilidade. Dependendo do alerta emitido através do *threshold* da *trigger*, os administradores da infraestrutura não podem demorar a saber do problema. Para isso, visando o provimento de alta disponibilidade também nesta emissão de alertas e garantir que os administradores saibam do problema, são necessários vários meios de transmissão deste alerta.

Existem diversas formas de se notificar os usuários do Zabbix (administradores da infraestrutura), porém o conceito é o de ser um método de troca de mensagens. Entre os mais viáveis, estão:

- Email: via SMTP no próprio servidor
- SMS: utilizando um celular conectado via USB ou serial
- *Jabber*: Protocolo utilizado para troca de mensagens instantâneas

- *Scripts* personalizados: Código para executar as tarefas acima e demais tarefas, desenvolvido pelo próprio administrador do Zabbix

Para a demanda da empresa, foram configurados alertas via: (i) Email; (ii) SMS; e (iii) *Google Talk* (*Script* personalizado (Apêndice A)). A configuração para envio do Email pode ser feita via interface gráfica, como mostra a Figura 4.5, bastando apenas ter o serviço SMTP habilitado no servidor ou um servidor de email apropriadamente configurado na rede – pode-se também utilizar um serviço de email como Gmail, Hotmail, Yahoo, etc.

Figura 4.5 – Adiciona mídia de email com configuração SMTP

4.3.8 Verificação de solidez do monitoramento

Com o monitoramento cumprindo seu papel, identificando falhas, alertando sobre picos de valores não esperados, esta atividade teve como objetivo reunir, novamente, os líderes dos projetos e gestores, a fim de avaliar a implantação do sistema. Ao longo destas reuniões, percebeu-se que aumentou a confiança interna no desenvolvimento do sistema como um todo e também a identificação prévia de falhas auxiliou no planejamento de prioridades a serem corrigidas.

Apresentou-se, então, o registro das atividades de todo o sistema, carga de processamento, memória, disco, identificou-se possíveis pontos de falha, onde a empresa deveria investir mais, elementos/serviços que não estavam alocados e o que necessitaria de um *upgrade*, entre outras questões. O cenário como um todo mostrou-se bastante produtivo e inovador para levantar questões que antes não eram observadas.

Além disso, o Zabbix também provê cruzamento de informações com visualização gráfica, ou seja, é possível reunir gráficos, sobrepostos, e visualizar em tempo real o que está acontecendo com aqueles itens daqueles ativos de rede. Ou até, uma falha que ocorreu de madrugada e foi identificada por um cliente que entrou em contato com o suporte. No mesmo momento desta falha pode-se pesquisar nos registros do Zabbix, filtrando pelos

ativos de rede que são relacionados à falha, e conseguir dados mais precisos ajudando a encontrar o erro. O que antes era ignorado até se encontrar um padrão e ser resolvido, agora pode ser identificado na hora e na primeira notificação encontra-se o problema.

4.3.9 Implementação do Zabbix Proxy

Após alguns dias com o Zabbix em funcionamento, sentiu-se uma sobrecarga muito alta sobre processador e disco rígido. Como são muitos dados coletados, a escrita é intensa, bem como a leitura, no momento de analisar e enviar alertas, por exemplo. Além do mais, como o Zabbix Server encontra-se em ponto geográfico diferente da rede que é monitorada (apesar de estar acessível via roteamento, na mesma rede lógica), este processamento de dados pode ser feito na própria rede (infraestrutura elétrica e de rede), sem prejudicar a confiabilidade do monitoramento, pois os alertas ficam a cargo do Zabbix Server. Sendo assim, decidiu-se implantar também o recurso de Proxy do Zabbix, instalado em outro servidor presente no Datacenter.

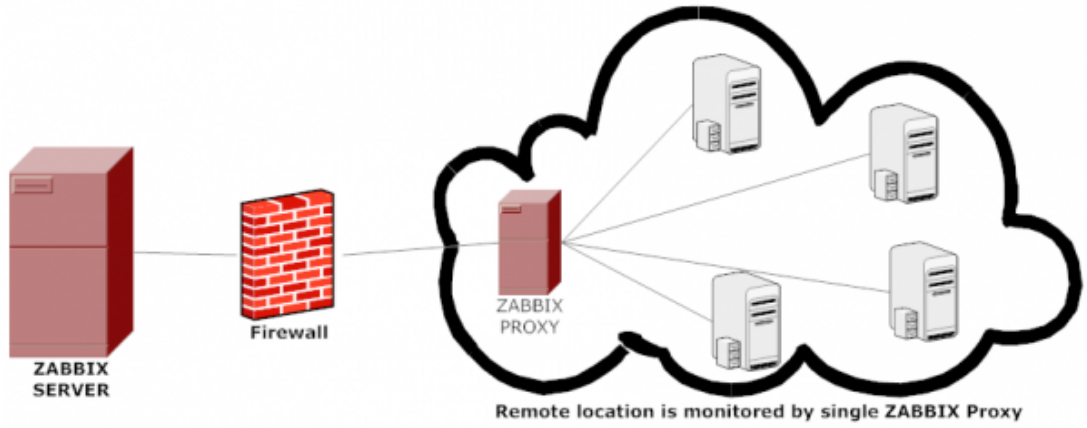

Figura 4.6 – Topologia genérica, como exemplo de uso do Zabbix Proxy

Como pode ser notado na Figura 4.6, o Zabbix Proxy fica responsável pela coleta dos dados e os encaminha ao Zabbix Server. Isso diminui um pouco da carga de processamento do Zabbix Server, uma vez que ele recebe dados já coletados e parcialmente analisados. Como existe, nesta topologia, apenas uma conexão TCP entre Zabbix Server e Zabbix Proxy, facilita também quando há um *firewall* e devem ser aceitos filtros para endereços IP e portas negociados pelo Zabbix. Outra vantagem é naturalizar (atingir outro objetivo, diferente do proposto) no ambiente uma replicação de dados, pois os dados coletados pelo Zabbix Proxy são armazenados em seu banco de dados local antes de serem transmitidos ao Zabbix Server, o que pode minimizar perda de dados em caso de uma falha na comunicação.

A instalação e configuração do Zabbix Proxy se dá da mesma forma que o Zabbix Server, bastando apenas alterar algumas *flags* de configuração (como recomendado na documentação oficial), como no Comando 4.6. No Zabbix Proxy pode ser utilizado o *sqlite*

ao invés de *mysql*, pela sua rápida configuração e também eficiência de acesso. Porém, se houver um crescimento exponencial de dados, pode se tornar difícil de gerenciar, uma vez que este gerenciador de banco de dados não é recomendado para grandes estruturas (vide o site do sqlite: <http://www.sqlite.org/features.html>).

```
1 # ./configure --prefix=/usr --enable-proxy --with-net-snmp --with-sqlite3
    --with-ssh2
```

Comando 4.6 – Configuração para compilação do Zabbix Proxy

Feita a instalação do Zabbix Proxy, é necessário configurá-lo no Zabbix Server, de modo que este saiba que existe um Proxy para seus serviços na rede. Após adicionar o Zabbix Proxy, basta mudar todos os ativos de rede, no Zabbix Server, para serem monitorados via Proxy. Desta forma, os dados serão coletados pelo Proxy.

Como conclusão da implantação do Proxy, houve uma pequena melhoria de processamento no Zabbix Server, porém a taxa de leitura/escrita em disco deste permanece muito alta. Acredita-se ser um gargalo do hardware (disco), uma vez que o Zabbix Server encontra-se instalado em uma máquina com configurações de Computador Pessoal.

4.4 Considerações Finais

Neste capítulo apresentou-se detalhadamente as atividades realizadas no período de estágio. Como descritas, as atividades envolvem conceitos intermediários de redes de computadores, sistemas operacionais, lógica de programação, banco de dados, entre outros. As atividades foram desenvolvidas com reuniões diárias e tomadas de decisões baseadas em discussões profundas sobre a demanda da empresa e o conhecimento necessário para a execução.

5 Conclusão

5.1 Dificuldades encontradas

Durante o período de estágio, a área principal de atuação foi a Infraestrutura. A empresa possui seu próprio Datacenter para que seus serviços não dependam de terceiros, isso facilitou a maioria das configurações necessárias. No entanto, a plataforma de produtos teve que ser cuidadosamente revisada, pelo fato dos servidores não estarem sendo monitorados. A migração do estado "não monitorado"para o estado "monitorado"causou certo alvoroço no início, mas ao mesmo tempo a empresa precisava desta implantação de monitoramento.

Poucas dificuldades sérias foram encontradas neste trabalho, uma vez que a empresa dá total liberdade e deposita extrema confiança nos profissionais desta área, sendo os superiores atenciosos sempre que necessário. Como é uma empresa que trata de dados financeiros de terceiros, muitas informações são confidenciais e precisou de uma conversa mais longa para envolver o monitoramento no meio do processo.

Com relação à parte técnica, foi sentida uma dificuldade por parte do estagiário em monitorar certos serviços que possuíam restrições em serem monitorados, *i.e.*, serviços que não são diretamente atrelados ao protocolo SNMP e o sistema operacional onde eles são executados não suportam a instalação do agente do Zabbix. Foram necessários vários conceitos aprendidos ao longo do curso para que fosse contornado este problema e a solução entregue como solicitado. Um dos grandes problemas enfrentados foi que, apesar de o Zabbix integrar-se à grande maioria das plataformas, através de protocolos conhecidos e execução de *scripts*, muitas demandas solicitadas pela empresa eram de difícil acesso e, portanto, recolhimento de dados para geração de gráficos e análises mais profundas.

As atividades mais dispendiosas, com relação à tempo e complexidade, foram a configuração das *triggers*, *thresholds* para estas *triggers* e a integração com sistemas de difícil acesso, como *switches* gerenciáveis. Tais tarefas necessitaram grande estudo em protocolos, como SNMP, HTTP, TCP/IP, BGP, e também a utilização de algum tipo de lógica para a resolução de problemas.

5.2 Contribuição para Formação

O principal conhecimento utilizado na execução das tarefas foi adquirido na disciplina de Redes de Computadores. Foram postos em prática todos os conceitos, desde o básico do protocolo TCP/IP, até roteamento avançado utilizando protocolos OSPF e

BGP. Acredita-se que sem os estudos da disciplina de Redes de Computadores, teria-se gasto um tempo extra em livros técnicos sobre o assunto.

Além de Redes, necessitou-se também um conhecimento do sistema Linux, adquirido na disciplina de Sistemas Operacionais. Apesar de não muito citado no relatório de estágio, por várias vezes fez-se necessário avaliar, por exemplo, a taxa de escrita/leitura em disco de um servidor, ou mesmo avaliar entre servidores com processador, memória e disco variáveis, tendo que tomar decisões baseadas nos números coletados. Alguns discos foram reformatados e testados com outros sistemas de arquivos, para avaliar a troca do sistema de arquivos padrão. Foi aplicado também conhecimento de virtualização, recursos compartilhados, entre outros.

Para aplicar este conhecimento em Sistemas Operacionais, utilizou-se a disciplina de Administração de Serviços de Redes de Computadores, onde pôde-se aprender a manusear o sistema Linux, instalar e configurar serviços locais e remotos, configuração de serviços em máquinas virtuais, configurar uma conexão entre máquina virtual e máquina física, utilizando a interface virtualizada de rede criada no sistema operacional hospedeiro, entre outros.

Utilizou-se também conhecimentos em programação e matemática, provenientes dos cursos iniciais, como Algoritmos e Estruturas de dados, Técnicas de Programação, Teoria da Computação e Matemática Discreta, bem como Cálculo e Física. Alguns deles não sendo utilizados diretamente.

Em paralelo, não apenas o curso de Ciência da Computação possibilitou aplicar conhecimentos e atravessar atalhos por fornecer conhecimento teórico, mas também o estágio proporcionou o discernimento de um sistema (produto) por completo, desde sua importância técnica internamente para a empresa até sua importância para cada cliente que depende deste sistema. Para uma empresa que possui dezenas de milhares de clientes utilizando um serviço oferecido na internet, que funciona 24/7/365 (24 horas por dia, 365 dias por ano), é fundamental manter uma rede altamente disponível, estável, rápida e sem falhas.

Do ponto de vista gerencial, como se estuda em Governança de TI, o levantamento de requisitos dos clientes e o *feedback* dos mesmos é um ponto-chave para uma empresa manter-se no mercado, tanto *online* quanto *offline*. Isso se deve a alta concorrência, fator este que foi notado mesmo na área de Infraestrutura, que lida pouco com clientes. Manter uma boa relação com clientes e parceiros é o primeiro passo a ser tomado por um gestor de empresas. Observar esta importância é ainda mais importante.

Enfim, para a área de Infraestrutura é necessário conhecimento em todo o curso de Ciência da Computação, com ênfase em hardware e conhecimentos também em elétrica. Gerenciar e identificar falhas em um Datacenter, ou numa rede, por menor que seja, implica

em desenvolvimento de soluções completas, que envolvam uma vasta base de conhecimento.

5.3 Considerações Finais

Este trabalho teve como objetivo a melhoria na identificação e prevenção de falhas na infraestrutura da empresa Gerencianet. Para isso, implantou-se um sistema de monitoramento bastante utilizado neste mercado, o Zabbix. Este sistema, dentre os estudados, mostrou-se como o mais adequado sistema em termos de tempo de aprendizado, tempo de configuração e instalação e facilidade na gerência de toda a rede.

Foi implementada uma solução (com implementação leia-se: conhecimento e levantamento da rede e do sistema; estudos teóricos e práticos; apresentação de relatórios de previsão; implantação do sistema Zabbix; e apresentação de relatórios de mudanças. Estas tarefas caracterizam uma solução que foi implementada para a rede da empresa) que permite monitorar os ativos de rede cruciais para manter a alta disponibilidade dos serviços prestados pela empresa, bem como alguns outros ativos de rede importantes para o funcionamento interno dos trâmites empresariais, como desenvolvimento, administrativo e suporte. Restam, ainda, o monitoramento de alguns equipamentos de rede, como *switches* e equipamentos que fazem a interligação das sedes, visando a obter a mesma qualidade de serviço que se tem com relação a prestação de serviço aos clientes.

Como o Zabbix é altamente personalizável, as sugestões surgem dos monitoramentos baseados em situações específicas, ou seja, se uma tabela no banco de dados MySQL depende de outra tabela para seu funcionamento, mas a primeira está corrompida, realizar este tipo de monitoramento e sugerir diretamente à aplicação para que realize os devidos procedimentos para que os clientes não percebam a falha. São ajustes finos necessários para a garantia de qualidade de todo o serviço prestado externa e internamente.

Com relação ao estágio, a empresa, representada pelos seus gestores, se mostrou aberta a ideias e críticas durante o período da execução das tarefas. Os envolvidos diretamente no projeto se mostraram interessados e disponíveis sempre que necessário. A empresa sempre disponibilizou um ambiente amigável, seguro e confortável para a execução das tarefas, o que foi primordial para que elas fossem finalizadas com sucesso.

Por fim, com o monitoramento da rede e dos serviços que nela operam, os administradores da rede tem completa gerência sobre a mesma, podendo identificar e prever falhas, gerando precisos relatórios de instabilidades para os gestores, bem como auxiliar em casos em que, para a rede não é problema, mas para um ponto específico do sistema causou algum dano a algum cliente ou a alguma tarefa que opera na rede na camada de aplicação.

Referências

ABBATE, J. E. From arpanet to internet: A history of arpa-sponsored computer networks, 1966–1988. 1994. Citado na página 19.

AMARAL, K. T. V. d.; DIAS, S. R. Uma proposta para documentação de redes. *Anuário da Produção de Iniciação Científica Discente*, v. 13, n. 16, p. 303–315, 2011. Citado na página 18.

BRESSOUD, T. C.; SCHNEIDER, F. B. Hypervisor-based fault tolerance. In: ACM. *ACM SIGOPS Operating Systems Review*. [S.l.], 1995. v. 29, n. 5, p. 1–11. Citado na página 28.

CACERES, R. et al. Measurement and analysis of ip network usage and behavior. *Communications Magazine, IEEE*, v. 38, n. 5, p. 144–151, May 2000. ISSN 0163-6804. Citado na página 21.

CARNEIRO, G.; FORTUNA, P.; RICARDO, M. Flowmonitor: A network monitoring framework for the network simulator 3 (ns-3). In: *Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools*. ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009. (VALUETOOLS '09), p. 1:1–1:10. ISBN 978-963-9799-70-7. Disponível em: <http://dx.doi.org/10.4108/ICST.VALUETOOLS2009.7493>. Citado na página 18.

CASE, J. et al. *Introduction and Applicability Statements for Internet-Standard Management Framework*. IETF, 2002. RFC 3410 (Informational). (Request for Comments, 3410). Disponível em: <http://www.ietf.org/rfc/rfc3410.txt>. Citado na página 21.

DIAS, H. *A importância do monitoramento de ativos de redes: um estudo de caso com o sistema Cacic*. 2008. Citado na página 21.

DRILLING, T. *Zabbix Release 2.2 - A Closer Look*. 2014. Disponível em: <http://www.admin-magazine.com/Archive/2014/20/Zabbix-release-2.2>. Citado na página 18.

ELLANTI, M. *Next Generation Transport Networks: Data, Management, and Control Planes*. [S.l.]: Springer, 2005. ISBN 9780387240671. Citado na página 21.

ERICSON, E. C.; ERICSON, L. T.; MINOLI, D. Book. *Expert systems applications in integrated network management / Eric C. Ericson, Lisa Traeger Ericson, and Daniel Minoli, editors*. [S.l.]: Artech House Norwood, MA, 1989. xii, 451 p. : p. ISBN 0890063788 0890063788. Citado na página 22.

GASPARY, L. et al. Uma arquitetura para gerenciamento distribuído e flexível de protocolos de alto nível e serviços de rede. *IXX Simpósio Brasileiro de Redes de Computadores*, 2001. Citado na página 21.

GLITHO, R. H. Contrasting osi systems management to snmp andtmn. *J. Netw. Syst. Manage.*, Plenum Press, New York, NY, USA, v. 6, n. 2, p. 113–133, jun. 1998. ISSN

1064-7570. Disponível em: <http://dx.doi.org/10.1023/A:1018754624498>. Citado na página 20.

HAWKINSON, J.; BATES, T. *RFC 1930: Guidelines for creation, selection, and registration of an Autonomous System (AS)*. 1996. Disponível em: <ftp://ftp.internic.net/rfc/bcp6.txt,ftp://ftp.internic.net/rfc/rfc1930.txt,ftp://ftp.math.utah.edu/pub/rfc/bcp6.txt,ftp://ftp.math.utah.edu/pub/rfc/rfc1930.txt>. Citado na página 17.

JUNIOR, M. L. P.; ROCHOL, J. Uma ferramenta para auxílio no gerenciamento de redes com backbone atm. *Porto Alegre: CPGCC da UFRGS*, 1998. Citado na página 21.

KUROSE, J. F.; ROSS, K. W. *Redes de Computadores e a Internet: Uma abordagem top-down*. Trad. 5 ed. São Paulo: Addison Wesley, 2010. Citado 5 vezes nas páginas 7, 18, 19, 20 e 25.

LAI, K.; BAKER, M. *Measuring Bandwidth*. 1999. Citado na página 21.

LINFO, T. L. I. P. *Shell Definition*. 2006. Disponível em: <http://www.linfo.org/shell.html>. Citado na página 32.

LOPES, R. P.; OLIVEIRA, J. L. An approach to self management based on automatic diagnostics. 1997. Citado na página 21.

MELCHIORS, C. *Raciocinio baseado em casos aplicado ao gerenciamento de falhas em redes de computadores*. Tese (Doutorado) — UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL, 1999. Citado na página 22.

MENDES, D. R. *Redes de Computadores*. [S.l.]: São Paulo: Novatec, 2007. Citado na página 18.

PISCITELLO, D. M.; CHAPIN, A. L. *Open Systems Networking: TCP/IP and OSI*. 1st. ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1993. ISBN 0201563347. Citado na página 20.

ROSE, M. T. *The Simple Book: An Introduction to Networking Management: Revised Second Edition*. 2nd. ed. [S.l.]: Simon & Schuster Trade, 1996. ISBN 0134516591. Citado na página 21.

ROSEN, E. C. *RFC 789: Vulnerabilities of network control protocols: An example*. 1981. Status: UNKNOWN. Not online. Disponível em: <ftp://ftp.internic.net/rfc/rfc789.txt,ftp://ftp.math.utah.edu/pub/rfc/rfc789.txt>. Citado na página 19.

SAYDAM, T.; MAGEDANZ, T. From networks and network management into service and service management. *Journal of Network and Systems Management*, Kluwer Academic Publishers-Plenum Publishers, v. 4, n. 4, p. 345–348, 1996. ISSN 1064-7570. Disponível em: <http://dx.doi.org/10.1007/BF02283158>. Citado na página 19.

STALLINGS, W. *SNMP, SNMPv2, and CMIP: the practical guide to network-management standards*. Addison-Wesley, 1993. ISBN 9780201633313. Disponível em: <http://books.google.com.br/books?id=2_hSAAAAMAAJ>. Citado na página 20.

STALLINGS, W. *SNMP,SNMPV2,Snmpv3,and RMON 1 and 2*. 3rd. ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1998. ISBN 0201485346. Citado na página 21.

Anexos

ANEXO A – *Script* em Python para integração com Google Talk

```
#!/usr/bin/python -W ignore::DeprecationWarning
import sys, os, xmpp, getopt, syslog

def main(argv):
    _debug = 0
  login = "usuario@gmail.com"
  pwd = "senha"

  if len(sys.argv) < 3:
    usage()
    sys.exit(2)

  rcptto=None
  subject=None
  msg=None

  rcptto=sys.argv[1]
  subject=sys.argv[2]
  msg=sys.argv[3]
  msg.replace(".","-")
  subject.replace(".","-")

  if subject != None and msg == None:
    msg = subject;
    subject = None;

  if rcptto == None or msg == None:
    usage()
    sys.exit(2)

    # Registra a mensagem no log (para debugs)
    log(msg)

    # Prepara a conexao com o Google
  def presenceHandler(conn, presence):
    if presence:
      if presence.getType() == "subscribe":
        cl.PresenceManager.ApproveSubscriptionRequest(pres.From)

  login=xmpp.protocol.JID(login)
```

```
41
42    if _debug == 1:
43      cl=xmpp.Client(login.getDomain())
44    else:
45      cl=xmpp.Client(login.getDomain(),debug=[])
46
47    # Conecta ao servidor do Google na porta padrao do protocolo xmpp (se
        falhar, sai)
48    if cl.connect( server=('google.com',5222)  ) == "":
49        sys.exit(0)
50
51    # Autenticacao (se falhar, sai)
52    if cl.auth(login.getNode(),pwd) == None:
53        sys.exit(0)
54
55    # Adiciona na conversa
56    pres = xmpp.Presence(to=rcptto, typ='subscribe')
57    cl.send(pres)
58
59    # Envia mensagem
60    cl.send(xmpp.protocol.Message(rcptto, msg, "chat"))
61    cl.disconnect()
62
63 # Definicoes de uso do script
64 def usage():
65    print "Usage:  {-d} [to] [subject] [body]"
66    print ""
67    print "Options:"
68    print "   [to]      email do destinatario"
69    print "   [subject]  assunto da mensagem"
70    print "   [body]     corpo do texto"
71
72 def log(text):
73    syslog.syslog(syslog.LOG_ERR, text)
74
75 if __name__ == "__main__":
76    main(sys.argv[1:])
```

Comando A.1 – gtalk.py (Adaptado de: <http://www.helviojunior.com.br/it/devel/enviando-alerta-do-zabbix-via-gtalk/>)