

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E  
TECNOLOGIA DE SÃO PAULO – CÂMPUS CAPIVARI**

**WASHINGTON ERNANDO PEREIRA BENÍCIO**

**MONITORAMENTO E GERENCIAMENTO DE REDES  
UTILIZANDO ZABBIX**

**CAPIVARI - SP**

**2015**

**WASHINGTON ERNANDO PEREIRA BENÍCIO**

**MONITORAMENTO E GERENCIAMENTO DE REDES  
UTILIZANDO ZABBIX**

Trabalho apresentado ao Curso de Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – Câmpus Capivari como requisito para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Alexandre Garcia Aguado

Coorientador: Edivaldo Serafim

**CAPIVARI - SP**

**2015**

WASHINGTON ERNANDO PEREIRA BENÍCIO

## **MONITORAMENTO E GERENCIAMENTO DE REDES UTILIZANDO ZABBIX**

Esta monografia foi julgada adequada à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas e aprovada em sua forma final pelo Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – Câmpus Capivari.

Apresentado à banca examinadora integrada pelos professores, em novembro de 2015.

---

Prof. Ms. Alexandre Garcia Aguado  
Orientador

---

Prof. Ms. Edivaldo Serafim  
Coorientador

---

Prof. Esp. Karlan Ricomini Alves  
Membro da banca

**CAPIVARI - SP**  
**2015**

## **DEDICATÓRIA**

Dedico este trabalho a minha mãe Maria José e meu pai Hilário pelo incentivo, minhas irmãs e irmão que de alguma forma contribuíram com este trabalho, minha esposa Patrícia pela força e dedicação comigo, minha filha Tamara pela motivação de meus esforços.  
Vocês são minha vida.

## **AGRADECIMENTO**

Primeiramente agradeço a Deus pela vida que me foi concedida com saúde e força de vontade.

Agradeço imensamente ao meu Professor orientador Alexandre Garcia Aguado por aceitar esse desafio, pela grande contribuição no desenvolvimento não só deste trabalho como também de minha formação profissional e pessoal.

Ao professor coorientador Edivaldo Serafim pela grande contribuição nas correções textuais deste trabalho e por todo o conhecimento técnico transmitido.

A equipe de técnicos de TI do câmpus Capivari pela prestatividade e ajuda no desenvolvimento deste trabalho.

Aos amigos e colegas de classe que de alguma forma contribuíram não só com este trabalho como também em outros de várias disciplinas.

*“Loucura é querer resultados diferentes fazendo tudo exatamente igual”.*

(Albert Einstein)

## RESUMO

Monitorar uma rede é verificar a eficácia do funcionamento de cada serviço, equipamento e processos existentes em uma mesma infraestrutura, seja ela empresarial ou residencial. O monitoramento de uma rede de computadores torna-se uma atividade essencial para, assim, garantir o seu funcionamento contínuo como também para assegurar um elevado grau de qualidade dos serviços oferecidos.

Devido a constante expansão do uso das redes de computadores, aumentam também os problemas, tais como: indisponibilidade de aplicação e/ou serviço, servidor de rede com baixa capacidade de processamento, entre outros.

Diante disso, o monitoramento em tempo real da infraestrutura de rede e seus ativos, vêm se tornando indispensável na gestão da tecnologia da informação. Esse monitoramento permite obter de modo rápido, preciso e confiável as informações necessárias sobre esses equipamentos, facilitando as tomadas de decisões no momento do planejamento, adequação e expansão do ambiente de tecnologia.

Com isso, propõe-se implementar uma arquitetura baseada em um modelo Gerente-Agente para permitir a automação da coleta dos dados de diversos componentes de uma rede de computadores, visando ampliar a aplicação das métricas e auxiliar no gerenciamento e monitoramento.

Uma ferramenta que ofereça e efetue essa coleta automatizada dos dados será implantada nas redes de computadores do Câmpus Capivari. Dentre as ferramentas existentes com estas características, a escolhida para essa tarefa é o Zabbix devido o licenciamento GPLv2, a gama de sistemas operacionais que é possível implantar os agentes e as diversas formas de notificações de alertas escalonadas.

**Palavras-chave:** monitoramento, gerenciamento, rede de computadores, tempo real, ativos de rede, gerente-agente, métrica.

## ABSTRACT

Monitor a network is to verify the effective operation of each service, equipment and existing processes in the same infrastructure, whether business or residential. The monitoring of a computer network becomes an essential activity to thus ensure its continued operation as well as to ensure a high level of quality of services offered.

Due to constant expansion of the use of computer networks, also increase the problems, such as unavailability of application and / or service network server with low processing power, among others.

Thus, the real-time monitoring of network infrastructure and its assets have become indispensable in the management of information technology. This monitoring allows for fast, precise and reliable information needed on such equipment, facilitating decision-making at the planning, adaptation and expansion of the technology environment.

With this, it is proposed to implement an architecture based on a Manager-agent model to allow the automation of data collection of various components of a computer network, aiming to expand the application of metrics and assist in management and monitoring.

A tool that offers and make this automated data collection will be located in the Campus Capivari computer networks. Among the existing tools with these features, chosen for this task is Zabbix because the GPLv2 license, the range of operating systems that you can deploy the agents and the various forms of staggered alert notifications.

**Keywords:** monitoring, management, computer network, real-time, network assets, managing agent, metric.



## Lista de Figuras

Figura 3.1: Camadas da arquitetura TCP/IP e alguns protocolos.....	17
Figura 3.2: Funcionamento da Camada de Aplicação.....	18
Figura 5.1: Principais componentes da arquitetura SNMP.....	24
Figura 5.2: Modelo de Comunicação entre Entidade Gerenciadora e Dispositivo Gerenciado.....	26
Figura 5.3: Árvore de identificadores de objetos.....	27
Figura 5.4: Árvore de OID's - Mib de impressora.....	28
Figura 6.1: Principais componentes do Zabbix trabalhando de forma separada.....	32
Figura 6.2: Zabbix Server unificado com o Banco de Dados e a Interface Web.....	33
Figura 6.3: Zabbix Proxy recebe os dados e envia ao Zabbix Server.....	33
Figura 7.1: Estrutura de Rede do Câmpus Capivari.....	39
Figura 7.2: Controle de cópias/impressões do Câmpus Capivari.....	42
Figura 7.3: Itens relevantes para o monitoramento de impressoras Sharp.....	44
Figura 7.4: Configuração para coletar o nível atual de t��ner da impressora.....	45
Figura 7.5: Impressoras monitoradas - Quantidade de impress��o e n��vel de t��ner.....	46
Figura 7.6: Configura��o m��nima do Zabbix Agent.....	48
Figura 7.7: Prot��tipo de item que captura o espa��o de HD.....	49
Figura 7.8: Prot��tipo de triggers para espa��o dispon��vel em HD.....	50
Figura 7.9: Gr��ficos das parti���es do servidor.....	50
Figura 7.10: Dashboard do PfSense em funcionamento no c��mpus Capivari.....	52
Figura 7.11: Tela de autentica��o para uso da rede Wi-Fi do c��mpus Capivari.....	53
Figura 7.12: Template Pfsense com suas aplica���es.....	54
Figura 1: Tela de boas vindas a instala��o do Zabbix Server.....	66
Figura 2: Tela de checagem de par��metros e depend��ncias de pacotes.....	66
Figura 3: Tela de configura��o da conex��o com banco de dados.....	67
Figura 4: Tela de teste de conex��o para habilitar pr��xima tela.....	67
Figura 5: Tela para defini��o da porta e nome do servidor.....	68
Figura 6: Tela para checagem final dos par��metros de configura��o.....	68
Figura 7: Tela de finaliza��o de instala��o.....	69

## Lista de tabelas

Tabela 1: Comunicação entre agente gerenciado e Zabbix <i>Server</i> .....	34
Tabela 2: Equipamentos existentes em junho de 2013.....	37
Tabela 3: Equipamentos existentes no Câmpus Capivari em Junho de 2015.....	38
Tabela 4: Servidores Físicos que atende as redes administrativa e laboratório.....	39
Tabela 5: Especificação do servidor virtual com Zabbix <i>Server</i> .....	40

# SIGLAS

API	<i>Application Program Interface</i>
ARP	<i>Address Resolution Protocol</i>
ARPA	<i>Advanced Research Projects Agency</i>
BD	<i>Banco de Dados</i>
CRE	<i>Coordenadoria de Registros Escolares</i>
DARPA	<i>Defense Advanced Research Projects Agency</i>
DHCP	<i>Dynamic Host Configuration Protocol</i>
FTP	<i>File Transfer Protocol</i>
GNU	<i>General Public License</i>
GSM	<i>Global System for Mobile Communication</i>
HD	<i>Hard Disk</i>
HTTP	<i>HiperText Transfer Protocol</i>
ICMP	<i>Internet Control Message Protocol</i>
IETF	<i>Internet Engineering Task Force</i>
IMP	<i>Interface Message Process</i>
IPMI	<i>Intelligent Platform Management Interface</i>
IPv6	<i>Internet Protocol version 6</i>
ISO	<i>International Organization for Standardization</i>
ITU-T	<i>Telecommunication Standardization Sector</i>
JMX	<i>Java Management Extensions</i>
LAN	<i>Local Area Network</i>
LDAP	<i>Lightweight Directory Access Protocol</i>
LLD	<i>Low Level Discovery</i>
MIB	<i>Management Information Base</i>
MIT	<i>Massachusetts Institute of Technology</i>
NMS	<i>Network Management Systems</i>
OID	<i>Object Identifier</i>
PHP	<i>PHP: Hypertext Preprocessor</i>
RFC	<i>Request For Comment</i>
RNP	<i>Rede Nacional de Ensino e Pesquisa</i>
SGBD	<i>Sistema de Gerenciamento de Banco de Dados</i>
SMI	<i>Structure Management Information</i>
SMS	<i>Short Message Service</i>

SMTP	<i>Simple Mail Transfer Protocol</i>
SNA	<i>Systems Networks Architecture</i>
SNMP	<i>Simple Network Managent Protocol</i>
TCP/IP	<i>Transmission Control Protocol/ Internet Protocol</i>
UDP	<i>User Datagram Protocol</i>
URL	<i>Uniform Resource Locator</i>
WAN	<i>Wide Area Network</i>
XML	<i>eXtensible Markup Language</i>
XMPP	<i>Extensible Messaging and Presence Protocol</i>

# Sumário

<b>1 INTRODUÇÃO.....</b>	<b>12</b>
<b>2 METODOLOGIA UTILIZADA.....</b>	<b>14</b>
<b>3 REDES DE COMPUTADORES.....</b>	<b>15</b>
3.1 A HISTÓRIA DAS REDES DE COMPUTADORES.....	15
3.2 ARQUITETURA TCP/IP.....	16
3.2.1 Camadas de rede modelo TCP/IP.....	17
<b>4 CONCEITOS DE GERENCIAMENTO DE REDE.....</b>	<b>20</b>
4.1 GERENCIAMENTO DE REDE.....	20
4.2 OBJETIVOS DA GERÊNCIA DE REDE.....	21
4.3 ATIVOS GERENCIÁVEIS EM UMA REDE DE COMPUTADORES.....	21
<b>5 PROTOCOLO DE GERENCIAMENTO.....</b>	<b>23</b>
5.1 SNMP.....	23
5.1.1 Arquitetura do SNMP.....	23
5.1.2 Operações do Protocolo SNMP.....	25
5.1.3 <i>Management Information Base (MIB – Base de Informação de Gerenciamento)</i> .....	26
<b>6 SOFTWARE DE MONITORAMENTO DE REDE ZABBIX.....</b>	<b>29</b>
6.1 SOBRE O ZABBIX.....	29
6.1.1 A história.....	29
6.1.2 Características e funcionalidades do sistema.....	30
6.1.3 Componentes do zabbix.....	31
6.1.4 Arquitetura do Zabbix.....	32
6.1.5 Comunicação entre o Agente e o Zabbix Server.....	34
<b>7 IMPLANTAÇÃO DO ZABBIX NO IFSP CÂMPUS CAPIVARI.....</b>	<b>35</b>
7.1 CENÁRIO DE PESQUISA.....	36
7.1.1 Equipe de Suporte a T.I.....	36
7.1.2 Infraestrutura de T.I.....	36
7.2 IMPLEMENTAÇÃO DO ZABBIX.....	40
7.3 PROBLEMA 1: GERENCIAMENTO DE IMPRESSORAS.....	41
7.3.1 Pré implementação.....	41
7.3.2 Processo de Implementação do Zabbix.....	43
7.3.3 Benefícios pós implementação.....	46
7.4 PROBLEMA 2: MONITORAMENTO DE ESPAÇO EM DISCO DOS SERVIDORES.....	47
7.4.1 Pré implementação.....	47
7.4.2 Processo de Implementação do Zabbix.....	48
7.5 PROBLEMA 3: INDISPONIBILIDADE DO WI-FI (HOTSPOT) DO CÂMPUS.....	51
7.5.1 Pré implementação.....	51
7.5.2 Processo de Implementação do Zabbix.....	54
7.5.3 Benefícios pós implementação.....	55
<b>8 CONSIDERAÇÕES FINAIS.....</b>	<b>56</b>
8.1 DIFICULDADES ENCONTRADAS.....	57
8.2 TRABALHOS FUTUROS.....	58
<b>9 REFERÊNCIAS.....</b>	<b>59</b>
<b>APÊNDICE A – INSTALAÇÃO DO ZABBIX SERVER 2.4 COM POSTGRESQL 9.3.61</b>	<b>61</b>
<b>APÊNDICE B – CONCLUSÃO DA INSTALAÇÃO DO ZABBIX 2.4 VIA BROWSER.66</b>	<b>66</b>
<b>APÊNDICE C – INSTALAÇÃO DO ZABBIX AGENT 2.4.7 EM LINUX.....</b>	<b>70</b>

# 1 INTRODUÇÃO

A disponibilidade é o ato de algo ou alguém estar sempre pronto para atender algum propósito. Analisando as redes de computadores, a expectativa que se tem é de um serviço que está sempre ativo, disponível para o uso e nos atender quando requisitado. A disponibilidade, acompanhada da confiabilidade são atributos básicos da segurança da informação. Conforme a NBR 5462/1994, disponibilidade é:

*Capacidade de um item estar em condições de executar uma certa função em um dado instante ou durante um intervalo de tempo determinado, levando-se em conta os aspectos combinados de sua confiabilidade, manutenibilidade e suporte de manutenção, supondo que os recursos externos requeridos estejam assegurados.*

Também na mesma norma NBR 5462/1994, confiabilidade é:

*Capacidade de um item desempenhar uma função requerida sob condições especificadas, durante um dado intervalo de tempo.*

As redes de computadores foram concebidas inicialmente como um meio para compartilhar dispositivos e serviços tornando-as parte do cotidiano como, uma ferramenta que ofereça um conjunto de recursos imprescindíveis para os usuários.

Com o objetivo de planejar o crescimento das redes estruturadas, efetuar o monitoramento e proporcionar uma alta disponibilidade dos recursos da rede, surge à necessidade do gerenciamento de redes de computadores. Com o aumento significativo dessas redes torna-se cada vez mais trabalhoso e difícil o gerenciamento realizado somente por esforços humanos, então a adoção de ferramentas automatizadas faz-se necessária.

Pensando nisso, propõe-se nesse trabalho apresentar com destaque a ferramenta Zabbix, implantá-la nas redes de computadores do IFSP câmpus Capivari. Com isso facilitar os trabalhos do administrador de rede, com monitoramento em tempo real de alto desempenho em dezenas de servidores, máquinas virtuais e dispositivos de rede que podem ser monitorados simultaneamente. (ZABBIX SIA)

Considerando esses aspectos, esse trabalho tem como problemática central contribuir para a seguinte questão: “*Quais as características e benefícios da implementação do Zabbix em uma rede do Instituto Federal de São Paulo Câmpus Capivari?*”

O trabalho está estruturado da seguinte maneira:

No capítulo 3 é abordada a origem e a evolução das redes de computadores. Para que o gerenciamento e monitoramento de redes de computadores sejam compreendidos, é importante conhecer como e o porquê as redes foram concebidas. Também será tratado a arquitetura de rede e de camadas no modelo TCP/IP, o conceito de Cliente/Servidor e os principais protocolos de monitoramento de rede.

No capítulo 4 são apresentados os conceitos iniciais sobre o gerenciamento de redes, sobre o paradigma de gerenciamento e sobre as métricas para avaliação de desempenho. É de suma importância entender esses conceitos para melhor compreensão do funcionamento das infraestruturas de medição. Além disso, a apresentação das áreas funcionais do gerenciamento de redes fornece a base para a compreensão das funções de gerenciamento previstas que será apresentado nesse trabalho.

No capítulo 5 é apresentado o protocolo de gerenciamento de redes da arquitetura Internet TCP/IP, o protocolo SNMP que atua na camada de aplicação. Também é contextualizada a arquitetura e as operações desse protocolo. Para concluir esse capítulo é apresentado o item que possui as informações a ser monitorada, que é a MIB (*Management Information Base*).

No capítulo 6 é abordada a história, as características e funcionalidades, os componentes e a arquitetura do software Zabbix que tem o importante papel de auxiliar um administrador/gerente de rede de computadores na tomada de decisões.

No capítulo 7 é contextualizado o cenário de pesquisa, a implementação do Zabbix nas redes da instituição de ensino. Foram coletados os principais problemas que a equipe de TI do campus recebe e com o Zabbix diagnosticado esses problemas, possibilitando assim uma intervenção mais precisa e pontual para as tomadas de decisões.

Por fim, é apresentada as considerações finais com as dificuldades encontradas no decorrer deste trabalho e propostas para trabalhos futuros com a ferramenta.

## 2 METODOLOGIA UTILIZADA

Dentro do conceito de método científico foram abordados duas grandes áreas que são pesquisa bibliográfica e pesquisa experimental, de forma que ambas consigam parametrizar todo o desenvolvimento deste trabalho.

Com a realização das pesquisas bibliográficas relacionado a rede de computadores, gerenciamento de rede, protocolo de gerenciamento e Zabbix possibilitando assim construir uma base de conhecimento sobre a ferramenta de monitoramento, e tentar responder a pergunta problema deste trabalho.

Já na pesquisa experimental é possível conhecer os procedimentos para o funcionamento do sistema, com isso contribui para a elaboração de manuais de instalação realizado a instalação do Zabbix Server no equipamento central, a instalação do Zabbix Agent em todos os servidores do câmpus e a habilitação do protocolo SNMP nos equipamentos a serem monitoras.

Através desta pesquisa é possível gerar análises quantitativa e qualitativa dos benefícios do sistema Zabbix implantado.

Na análise quantitativa são mensurados e controlados as quantidades de impressões e nível de t  ner dos equipamentos *outsourcing* de impress  o. Tamb  m analisa-se o n  vel de armazenamento nos HD's dos servidores e externos, com isso demonstrar a import  ncia da aquisi  o de mais unidades de armazenamento externo para os *backups*.

Com a an  lise qualitativa    poss  vel perceber o quanto diminuiu o tempo de resposta da equipe de TI do c  mpus Capivari e o *downtime* dos equipamentos, servi  os e sistemas.



## 3 REDES DE COMPUTADORES

### 3.1 A história das redes de computadores

Tudo começou por volta da década de 1960, com advento das redes telefônicas, que era o meio de comunicação dominante na época e utilizava o conceito de comutação de circuito para transmitir informações entre uma origem e um destino (KUROSE; ROSS, 2010. p. 45). Na mesma época o conceito da multiprogramação (*time-sharing*) estava sendo iniciado, tornando possível a interligação dos computadores geograficamente distribuídos para compartilhar informações.

Graças a três grupos de pesquisas independentes, considerados os precursores das redes de computadores são Leonard Kleinrock, que na época era discente (doutorado) do MIT (*Massachusetts Institute of Technology*), teve o primeiro trabalho publicado sobre técnica de comutação de pacotes. Paul Baran, do Rand Institute que iniciou uma investigação sobre a utilização de comutação de pacote na transmissão segura de voz pelas redes militares. E Donald Davie junto com Roger Scantlebury, da National Physical Laboratory, que desenvolviam suas ideias sobre esse assunto. (KUROSE, 2010. p. 45)

De fato, a primeira rede de computadores por comutação de pacotes que se sabe da história, foi implementado pela ARPA (*Advanced Research Projects Agency – Agência de Projetos de Pesquisa Avançada*), do Departamento de Defesa dos Estados Unidos, a ARPAnet. Em 1969 foi instalado o primeiro comutador de pacotes, o IMP (*Interface Message Process – Processador de Interface de Mensagem*) na Universidade da Califórnia em Los Angeles. A intenção era interligar os sistemas centrais de computadores de uma dúzia de universidades e instituições de pesquisa apoiadas financeiramente por essa agência. (DEITEL, 2005)

Em 1970, surgiram novas redes independentes de comutação de pacotes, porém eram redes isoladas, como por exemplo (KUROSE; ROSS, 2010, p. 46):

- ALOHAnet, era uma rede de micro-ondas que interligava universidades do Havaí, bem como as redes de pacote por satélite e por rádio;
- Telenet, uma rede de comutação de pacotes fundamentada na tecnologia da ARPAnet;
- Tymnet e GE *Information Service*, foram redes de tempo compartilhado;

- SNA da empresa IBM, cujo trabalho comparava-se ao da ARPAnet

Nota-se que o número de redes estava passando por uma transição em relação a quantidade e o crescimento muito rápido em um curto espaço de tempo, com isso foi de suma importância a DARPA (*Defense Advanced Research Projects Agency* – Agência de Projetos de Pesquisa Avançada de Defesa) patrocinar o desenvolvimento de uma arquitetura abrangente para conectar essas redes, nascendo ali o termo *internetting*, ou seja, rede das inter-redes (KUROSE; ROSS, 2010, p. 46).

Porém, para que de fato essas redes se comunicassem, houve a necessidade de padronizar as regras de comunicação entre aplicações (correio eletrônico, transferência de arquivo), digamos que, “com mesmo idioma” entre as redes. Assim, surgiu o protocolo TCP/IP (*Transmission Control Protocol* – Protocolo de Controle de Transmissão / *Internet Protocol* – Protocolo de Internet) – também patrocinado pela DARPA.

*“... As primeiras experiências com o TCP, combinadas com o reconhecimento da importância de um serviço de transporte fim a fim não confiável, sem controle de fluxo, para aplicações com voz em pacotes, levaram a separação entre IP e TCP e ao desenvolvimento do protocolo UDP ...” no final dos anos 1970 e já contava com aproximadamente 200 computadores conectados. (KUROSE; ROSS, 2010, p. 47)*

Oficialmente em 1 ° de Janeiro de 1983 o TCP/IP foi adotado como novo padrão para as máquinas da ARPAnet.

### 3.2 Arquitetura TCP/IP

A grande conquista da padronização do protocolo de comunicação das redes de computadores foi o TCP/IP, e este se deve a finalidade principal: conectar grandes redes de grande porte e também por ser roteável, ou seja, a possibilidade de um pacote de dados sair de sua origem até o destino por mais de uma rota ou caminho dentro da rede (TORRES, 2001, p. 64).

Outro fator muito importante que popularizou o uso do TCP/IP foi sua arquitetura aberta, ou seja, os fabricantes de equipamentos de rede podem implementar em seus equipamentos sem a necessidade de direitos autorais (TORRES, 2001, p. 64).

### 3.2.1 Camadas de rede modelo TCP/IP

A arquitetura TCP/IP divide o processo de comunicação entre computadores em uma hierarquia de quatro camadas ou pilhas de protocolos, sendo que cada uma tem o papel de inserir informações e repassa o pacote de dados para a camada adjacente. As informações contidas dentro do pacote só pode ser decodificadas pela mesma camada no computador receptor, ou seja, no destino da informação, no mesmo nível hierárquico.(COULOURIS, 2007, p. 92)

Essas camadas são divididas na seguinte ordem (DEITEL, 2005, p. 491):

- 4ª Camada – Aplicação
- 3ª Camada – Transporte
- 2ª Camada – Rede
- 1ª Camada – Interface de Enlace

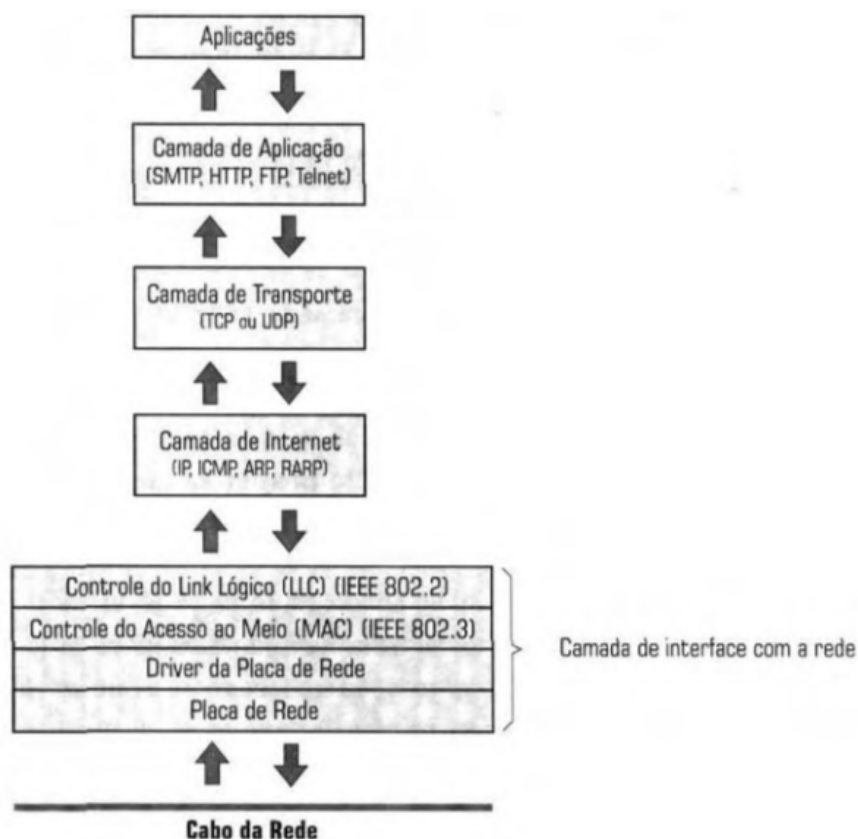


Figura 3.1: Camadas da arquitetura TCP/IP e alguns protocolos

Fonte: Gabriel Torres. Rede de Computadores, 2001

A Figura 3.1 mostra como esta estrutura é implementada e como ocorre o inter-relacionamento entre as camadas. Como se nota, cada camada presta serviços específicos para a rede, dando suporte para as camadas superiores da hierarquia.

A Camada de Aplicação é o nível mais alto da hierarquia das camadas e nela encontram-se alguns dos protocolos mais conhecidos como, HTTP (*HiperText Transfer Protocol*), SNMP (*Simple Network Managent Protocol*), SMTP (*Simple Mail Transfer Protocol*), FTP (*File Transfer Protocol*). Essa camada se comunica com a camada de transporte através das portas padrões. Exemplo quando um cliente de e-mail quer receber os e-mails armazenados no servidor de e-mail, o mesmo faz a requisição na camada de aplicação do TCP/IP, que é atendido pelo protocolo SMTP. (KUROSE; ROSS, 2010, p. 71)

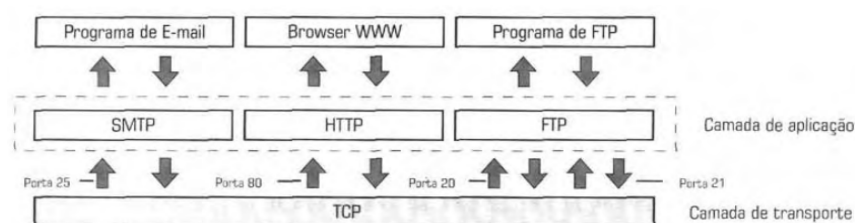


Figura 3.2: Funcionamento da Camada de Aplicação

Fonte: Gabriel Torres. *Rede de Computadores*, 2001

A camada de Transporte possui dois protocolos, o TCP (*Transmission Control Protocol*) e o UDP (*User Datagram Protocol*). Essa camada é responsável por receber os dados da camada de aplicação e quebrá-los em pacotes, anexando informações de controle em cada pacote e por fim encaminhá-lo para a camada de rede. Com a camada de aplicação, os pacotes são transmitidos de forma simultânea entre os dados de diversas aplicações, e isso só é possível graças ao conceito de portas, já que dentro do pacote há informação da porta de origem e de destino do dado do respectivo pacote, conforme ilustrado na Figura 3.2. (KUROSE; ROSS, 2010, pgs. 141-142)

Na camada de Rede podem existir vários protocolos que operam na rede como: IP (*Internet Protocol*); ICMP (*Internet Control Message Protocol*); ARP (*Address Resolution Protocol*). Conforme (KUROSE; ROSS, 2010, p. 229) essa camada é responsável por transportar pacotes de um *host* remetente a um *host* destinatário, através do roteamento essa camada determina a rota ou caminho que esses pacotes devem seguir da origem ao destino.

Na camada de Interface de Enlace um protocolo é responsável por movimentar *datagramas* de camada de rede nó a nó, que são os elementos da rede (roteadores) por um

único caminho, independente se esse caminho é através de meio elétrico – convertendo os dados binários em sinais elétricos –, por meio óptico – convertendo-os em sinais luminosos – ou por meio aéreo – convertendo em ondas eletromagnéticas. (KUROSE; ROSS, 2010, pgs. 319-322)

Portanto percebe-se que para haver a troca de informação entre dispositivos distintos são necessárias que algumas etapas entre as camadas sejam executadas e cada uma delas possuem seus protocolos específicos. Em uma aplicação de gerenciamento de rede um dos protocolos utilizados é o SNMP, o gerenciamento de redes é tratado no capítulo 4 e o protocolo específico de gerenciamento de rede pode ser visto com maiores esclarecimentos no capítulo 5.

## 4 CONCEITOS DE GERENCIAMENTO DE REDE

### 4.1 Gerenciamento de rede

Gerenciamento pode ser definido como coordenação, controle de atividade e monitoramento de recursos, assegurando na medida do possível confiabilidade, segurança e alta disponibilidade.

*“A área de gerência de rede foi impulsionada pela necessidade de monitoração e controle do universo de dispositivos que as redes de computação. Atualmente as redes de computadores e os recursos associados, além das aplicações distribuídas, tem se tornado fundamental e de tal importância para uma organização, que elas basicamente “não podem falhar”.”(LIMA, 1997, vol. 1, n. 3)*

Há uma analogia do conceito de gerenciamento de rede com uma cabine de avião. Nesse cenário, a cabine é um sistema complexo com muitos componentes que devem ser monitorados, gerenciados e controlados, já que é equipada com instrumentos para que o piloto possa ter o controle sobre os muitos componentes de uma aeronave. Na ocasião o piloto monitora equipamentos remotos e analisa os dados para garantir que os equipamentos estejam funcionando e operando dentro dos limites especificados. (KUROSE, ROSS, 2010, p. 553)

Existe uma definição muito interessante dita por SAYDAM relacionado base sólida de gerenciamento de rede:

*“Gerenciamento de rede inclui o oferecimento, a integração e a coordenação de elementos hardware, software, humanos, para monitorar, testar, consultar, configurar, analisar, avaliar e controlar os recursos de rede, e de elementos, para satisfazer às exigências operacionais, de desempenho e de qualidade de serviço em tempo real e um custo razoável.” (SAYDAM, 1996).*

Controlar reativamente um sistema fazendo ajustes de acordo com as modificações ocorridas em seu ambiente e gerenciar proativamente, possibilita detectar tendências ou anomalias que permitam executar ações antes que surjam problemas mais graves. Uma das funções do administrador de redes é justamente evitar ou responder de forma rápida a quaisquer tipos de anomalia. Para tal, deve-se possuir conhecimento suficiente e dispor de ferramentas que indique as falhas, e com isso, tomar atitudes necessárias a fim de manter a rede ativa e os serviços disponíveis.

## 4.2 Objetivos da gerência de rede

O gerenciamento está diretamente ligado ao controle das atividades e ao monitoramento dos recursos no ambiente a que se destina. Isso possibilita a otimização do uso dos recursos disponíveis e diminui o tempo de indisponibilidade do equipamento e/ou serviço de rede, principalmente subsidiando os gestores nas tomadas de decisões, por exemplo, *cluster* de servidor de arquivos, caso um equipamento pare de funcionar os gestores sejam alertados para definir uma estratégia rápida para resolver o incidente.

O foco para as atividades de gerenciamento de rede é a organização e, aspectos como o atendimento do usuário, se caracterizam como primordial para o sucesso da estrutura. É desejável que o usuário dos serviços de rede tenha um único ponto de contato para reportar problemas e mudanças. Os limites de atuação desta gerência devem levar em conta a amplitude desejada pelo modelo implantado na instalação e operação da rede.

O modelo de gerenciamento é dividido em três etapas, onde na primeira fase é **coletar os dados**, que consiste em um processo, em geral automatizado para obter os dados do recurso a ser gerenciado. Na segunda fase é o **diagnóstico**, que consiste no tratamento e análise dos dados coletados. O servidor (gerente) executa uma série de procedimento com o intuito de determinar a causa do problema representado no dispositivo gerenciado. E na última fase é a **ação ou controle**, uma vez diagnosticado o incidente cabe ao administrador executar uma ação ou controle sobre o recurso gerenciado (ONE LINEA TELECOM, 2015).

## 4.3 Ativos gerenciáveis em uma rede de computadores

Uma rede de computadores é uma infraestrutura de comunicação constituída por um conjunto de equipamentos interconectados, a fim de trocar e compartilhar recursos e informações entre si e prover uma gama de serviços ao usuário (TANENBAUM, 2003, p. 1).

Nesse sentido, qualquer equipamento e/ou serviço conectado a rede podem ser monitorados, por exemplo: servidores, *switchers*, roteadores, impressoras, *desktops*, suprimentos energéticos, etc, bem como os serviços de banco de dados (MySQL, PostgreSQL, SQL Server) e aplicações (Java, Tomcat, Jboss). Todo esse conjunto de elementos heterogêneos possui uma grande quantidade de dados que podem ser importantes principalmente para o gerenciamento de segurança da informação, conforme destaca (KUROSE; ROSS, 2010).

*“Um **dispositivo gerenciado** é um equipamento de rede, pode ser um hospedeiro, um roteador, uma ponte, um hub, uma impressora ou um*

*modem. No interior do dispositivo gerenciado pode haver diversos **objetos gerenciados**. Estes são, na verdade, as peças de hardware...e os conjuntos de parâmetro de configuração para as peças de hardware e software...Esses objetos gerenciados têm informações associadas a eles que são coletadas dentro de uma **Base de Informações de Gerenciamento** MIB (Management Information Base).” (KUROSE; ROSS, 2010, p. 558, grifos do autor)*

Dependendo da ênfase atribuída aos investimentos realizados no ambiente de rede, as funções de gerência devem ser centralizadas nos servidores. Esta estrutura pode definir aspectos como: a estratégia empregada no atendimento/chamadas dos usuários, atuação do pessoal envolvido nas tarefas de gerenciamento de rede, supridores de serviços, etc.

Para que haja a possibilidade de gerenciar os objetos de um dispositivo se faz necessário um protocolo de gerenciamento, pois é através dele que os parâmetros relevantes para tal fim foi implementado. Neste sentido o estudo desse protocolo consta no próximo item deste trabalho.



## 5 PROTOCOLO DE GERENCIAMENTO

### 5.1 SNMP

Como foi exposto no item 2.2.1, a arquitetura de rede trabalha com camadas e em cada uma existem seus protocolos de “conversação”. Nesse sentido, conforme (LIMA, 1997), o protocolo responsável por transportar informações de gerência de rede entre os dispositivos gerenciados e os sistemas de gestão de rede é o SNMP (*Simple Network Management Protocol* – Protocolo Simples de Gerenciamento de Rede).

“O protocolo SNMP refere-se a um conjunto de padrões para gerenciamento que inclui um protocolo, uma especificação de estrutura de dados, e um conjunto de objetos de dados...” (LIMA, 1997, vol. 1, n. 3)

Esse protocolo foi definido pela IETF (*Internet Engineering Task Force*) pelo qual as informações de gestão para um elemento de rede podem ser consultadas ou alteradas por usuários remotos. O protocolo foi projetado para ser independente de arquitetura ou plataforma (*hardware* ou *software*) ou de redes de fabricantes específicos, já que o protocolo foi projetado e oferecido conforme (RFC 1089). Em uma época em que a necessidade de gerenciamento de rede começava a ficar iminente, sendo que rapidamente aceito (KUROSE; ROSS, 2010, p. 558).

Desde então foram realizadas diversas atualizações neste protocolo e culminou no SNMPv1, conseqüentemente o SNMPv2 e chegou a versão mais recente, o SNMPv3 (RFC 3410) atualizado em dezembro de 2002.

“O SMNP é simplesmente um conjunto de operações e requisições que dá ao administrador a habilidade de obter e mudar informações e estados de dispositivos configurados para aceitar requisições SMNP.” (MOURA; SCHMIDT, 2005, p. 1)

#### 5.1.1 Arquitetura do SNMP

Desde então foi realizado diversas atualizações neste protocolo e culminou no SNMPv1, consExistem três componentes principais em uma arquitetura de gerenciamento de rede que são: entidade gerenciadora, dispositivos gerenciados e o protocolo de gerenciamento de rede ou dados, como mostra a Figura 5.1 (KUROSE; ROSS, 2010, p. 557).

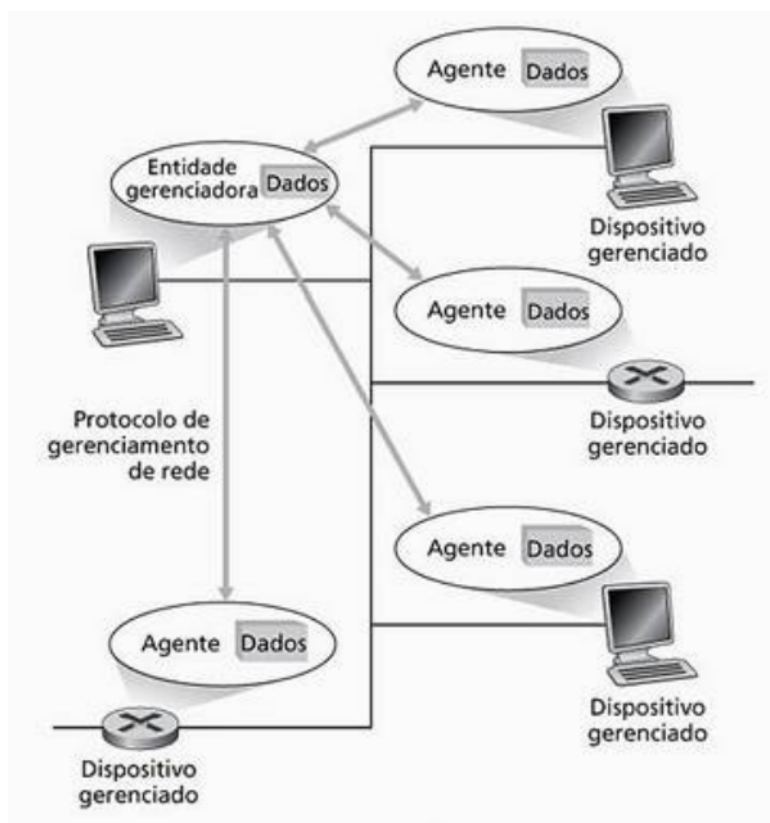


Figura 5.1: Principais componentes da arquitetura SNMP

Fonte: KUROSE; ROSS, 2010

A entidade gerenciadora é uma aplicação que em geral tem uma grande participação do administrador de rede. Comumente executado em uma estação central de gerência de rede, trata-se do local da atividade de gerenciamento. Tem o importante papel de controlar a coleta, o processamento e analisar as informações de gerenciamento de rede, ou seja, faz o intermédio entre as aplicações e os agentes espalhados pelos dispositivos.

A API (*Application Program Interface*) é um conjunto de funções que fazem o intermédio na execução de comandos entre um programa e outro, de forma a simplificar um programa a ter acesso a funcionalidades de outro programa e que, no caso do SNMP, faz o intermédio das execuções entre uma aplicação de gerência e o gerente SNMP (TANENBAUM, 2003, p. 558).

O dispositivo gerenciado conforme exposto no item 3.4, trata-se do equipamento de rede ou *software*, podendo ser um roteador, *switch*, impressora ou qualquer aplicação como banco de dados por exemplo. Dentro do dispositivo pode haver diversos objetos gerenciados (placa de interface de rede, nível de tóner de impressora, espaço em disco rígido, etc). Os

dispositivos gerenciados respondem a solicitações da entidade gerenciadora e executa ações demandas. Nos objetos gerenciados possuem informações associadas a eles que são coletados dentro de uma MIB.

O protocolo de gerenciamento (SNMP) é executado entre a entidade gerenciadora e o agente de gerenciamento dos dispositivos gerenciados. Esse protocolo pode ser usado para informar a entidade gerenciadora alguma anormalidade no dispositivo gerenciado. É importante lembrar que o protocolo de gerenciamento não gerencia a rede e sim fornece ferramenta com a qual o administrador de rede pode gerenciar, ele é usado para transmitir informações e comandos entre a entidade gerenciadora (gerente) e o dispositivo gerenciado (agente) (KUROSE; ROSS, 2010, p. 557).

### **5.1.2 Operações do Protocolo SNMP**

Existem três tipos de operações comuns durante o processo de gerenciamento:

*Get* (Obter): Permite que a entidade gerenciadora recupere o valor de objetos MIB do dispositivo gerenciado. Essa recuperação de valor pode ser de dois tipos: *GetRequest*, onde recupera a primeira informação da lista de informação de objetos ou *GetNextRequest*, onde recupera a próxima informação disponível na lista, a partir da última informação solicitada;

*Set* (Definir): Permite que a entidade gerenciadora defina o valor de objetos MIB do dispositivo gerenciado, para cada *SetRequest* executado pela entidade gerenciadora, o dispositivo gerenciado responde com um *GetResponse*;

*Trap* (Armadilha): permite que o dispositivo gerenciado notifique a entidade gerenciadora sobre eventos significativos.

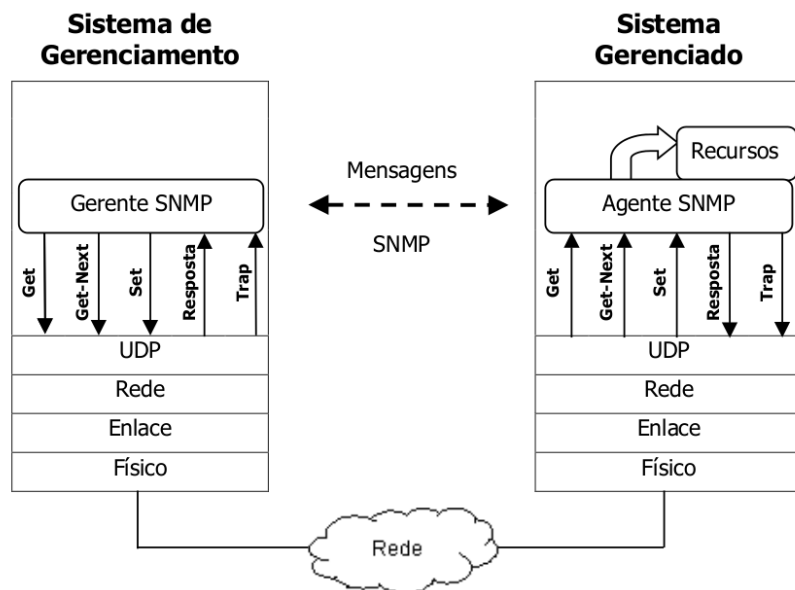


Figura 5.2: Modelo de Comunicação entre Entidade Gerenciadora e Dispositivo Gerenciado

Na Figura 5.2 é possível verificar a comunicação entre a entidade gerenciadora (Gerente) e o dispositivo gerenciado (Agente), a troca de mensagens ocorre na camada de aplicação e os principais parâmetros para haver essa interligação

### 5.1.3 Management Information Base (MIB – Base de Informação de Gerenciamento)

A estrutura das informações de gerenciamento (SMI – *Structure Management Information* – Estrutura de Informação de Gerenciamento) define o formato geral para a construção de uma MIB (*Management Information Base* – Base de Informação de Gerenciamento). A SMI identifica os tipos de dados que podem ser usados na MIB e especifica quais os recursos que serão representados e nomeados. (STALLINGS, 2005)

Conforme mostra a Figura 5.3, os objetos gerenciados são organizados de modo hierárquico, em estrutura de árvore, e no topo dessa hierarquia é apresentado as duas principais entidade de padronização ISO e o ITU-T. Cada item da árvore representa algum recurso ou alguma informação relacionada a um elemento gerenciado, cada objeto gerenciado precisa ter um identificador único OID (*Object Identifier* – Identificador de Objeto) que serve como nome para o objeto, as referências desses objetos podem ser numéricas ou nominal.

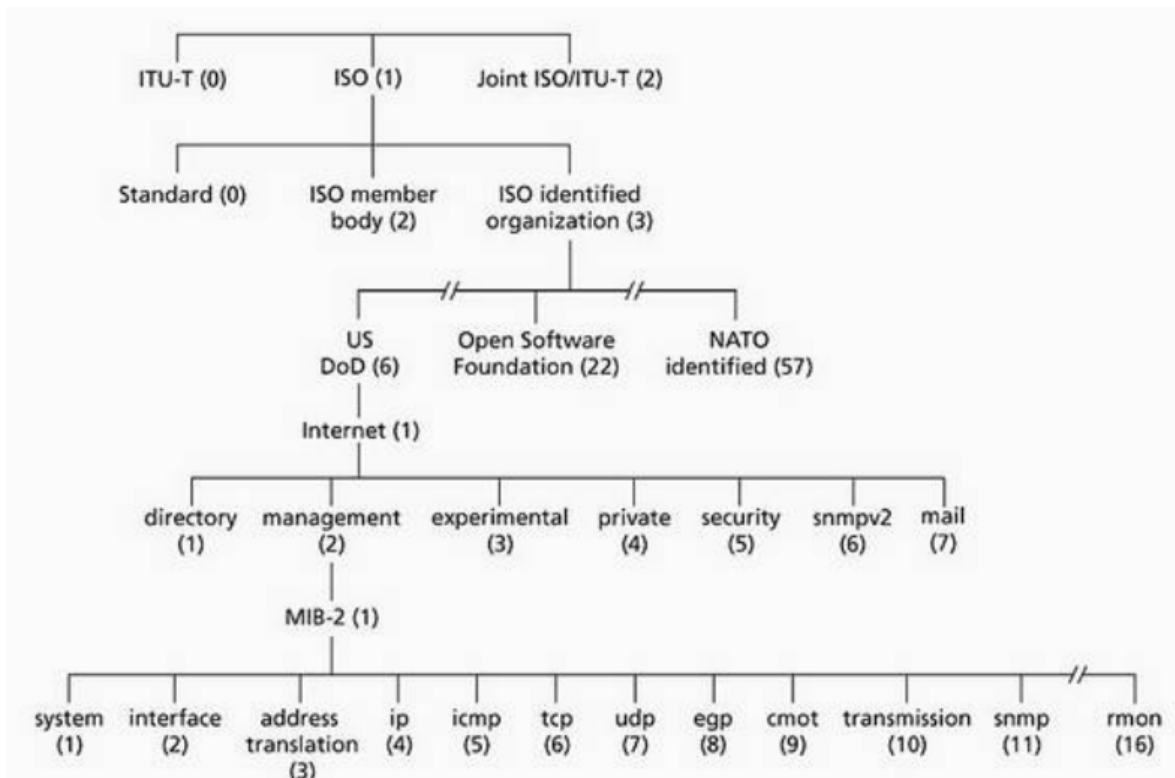


Figura 5.3: Árvore de identificadores de objetos

Fonte: KUROSE; ROSS, 2010

Para um melhor compreensão sobre a árvore de identificadores de objetos em um equipamento de rede, vamos utilizar como exemplo os dispositivos de impressão. O grupo que padroniza as impressoras é o OID 1.3.6.1.2.1.43 ou na forma nominal separado por um ponto {iso.identified-organization.dod.internet.mgmt.mib-2.**printmib**} conforme visto na Figura 5.4.



Figura 5.4: Árvore de OID's - Mib de impressora

Fonte: <http://oid-info.com/cgi-bin/display?tree=1.3.6.1.2.1.43>

## 6 SOFTWARE DE MONITORAMENTO DE REDE ZABBIX

### 6.1 Sobre o Zabbix

Conforme HORST; PIRES; DÉO (2015, p. 19) Zabbix é uma ferramenta moderna, *Open Source GPLv2 (GNU General Public License)* e multiplataforma. Possui apenas uma versão, que é considerada *Enterprise*, sendo utilizado para monitorar a disponibilidade e o desempenho de aplicações, ativos e serviços de rede.

Para a ZABBIX SIA é possível coletar diversos tipos de dados da rede, oferecendo grande performance para isso, e pode ser implementado em ambientes muito grandes. É um *software* de nível empresarial projetado para disponibilidade e desempenho de componentes de uma infraestrutura de TI.

#### 6.1.1 A história

O sistema Zabbix foi idealizado e projetado pelo Russo Alexei Vladishev em 1998, devido à necessidade de utilizar uma ferramenta de monitoramento em um grande Banco na Letônia, onde trabalhava como administrador de sistemas. Na ocasião as ferramentas existentes eram pagas e exigiam conhecimentos avançados. Apesar de já idealizado e projetado, o lançamento da primeira versão *release* do sistema foi ocorrer somente em 2001 e a primeira versão (V. 1.0) estável foi lançada em 2004 (ZABBIX, 2015).

A Zabbix SIA Company est., foi lançada em 2005 com a intenção de fornecer serviço de suporte técnico especializado e hoje a empresa possui escritórios nos Estados Unidos, Europa e Japão. Como o sistema é de código fonte aberto, várias comunidades de software livre espalhadas pelo mundo também colaboram com seu desenvolvimento (ZABBIX, 2015).

Na versão 1.1 lançada em 2006 foi disponibilizado vários recursos, entre eles, o uso de *Modem GSM* possibilitando assim o envio de alertas por SMS; Monitorar arquivos de *Log* possibilitando auditorias no sistema; Suporte aos protocolos SNMPv3 e *XML Protocol*. Em 2006 com a versão 1.4, houve uma reformulação na *Interface*, possibilitando ainda a utilização do Banco de dados SQLite; Possibilidade de descoberta de rede automática; Suporte a IPv6. No ano de 2008 com liberação da versão 1.6 foi disponibilizado método de autenticação usando LDAP; Na tela principal (*Dashboard*) os elementos dinâmicos; Implementação do Zabbix Proxy. Com a versão 1.8 lançada em 2009, implementou o suporte a IBM DB2, Zabbix Proxy de forma passiva, suporte ao uso de expressão regular, melhorias

na autodescoberta e suporte ao autorregistro (ZABBIX, 2015).

Em 2012 a ZABBIX SIA abre uma filial no Japão (Zabbix Japan LLC), também no mesmo ano foi disponibilizado a versão 2.0, com a possibilidade de monitoramento de Trap do SNMP, suporte a monitoramento remoto JMX (*Java Management Extensions*), inventário automático. No ano de 2013, foi liberada a versão 2.2 implementou o monitoramento da vCenter e vSphere adicionando suporte e monitoramento de desempenho a plataformas virtualizadas como Vmware e Hypervisors, suporte a macros de usuário e *scripts* globais, LLD (*Low Level Discovery* - (Descoberta de Baixo Nível) possibilitando assim uma forma automatizada para criar itens, *triggers* e gráficos em diferentes recursos do *hosts* (ZABBIX, 2015).

Em 2014 surgiu a versão 2.4 com melhoramentos significativos no monitoramento Web, também na sintaxe das expressões de *Trigger*, entre outras novidades. Lançada em agosto de 2015 a versão 3.0 possui uma nova interface pensada na usabilidade.

### **6.1.2 Características e funcionalidades do sistema**

O Zabbix suporta vários SGBD's (Sistema de Gerenciamento de Banco de Dados) para armazenamento dos dados de coletas e configurações. Os SGBD's suportados são: MySQL/Mariadb, PostgreSQL, SQLite, Oracle e IBM DB2. Os módulos de sincronismo e coleta de dados foram desenvolvidos na linguagem C e a interface web foi utilizado o PHP (PHP: **H**ypertext **P**reprocessor).

Segundo HORST; PIRES; DÉO (2015, p. 20), o Zabbix possui dezenas de módulos e suas principais funcionalidades estão listadas a seguir:

- Autodescoberta de dispositivos de rede;
- Autodescoberta de recurso do *host* (por verificação simples, agente ou SNMP);
- LLD que permite criar itens, triggers, gráficos para diferentes recursos do host;
- Monitoramento distribuído com administração centralizada via web por meio do uso de proxy;
- Aplicação servidor compatível com os sistemas operacionais: GNU/Linux, IBM AIX, Solaris, HP-UX, AIX, FreeBSD, NetBSD, OpenBSD, Mac OS X;
- Aplicação cliente de alto desempenho compatíveis com todos os sistemas operacionais da aplicação servidor com o incremento do Windows (edições Server e Desktop);



- Monitoramento com ou sem o uso de Agentes;
- Auditoria do sistema;
- Envio de alertas por e-mail, SMS, Jabber XMPP (*Extensible Messaging and Presence Protocol*) e scripts personalizados;
- Monitoramento de aplicações Java;
- Monitoramento de dispositivos via IPMI (*Intelligent Platform Management Interface*);
- Monitoramento de aplicações Web;
- Monitoramento de ambientes virtualizados.

### 6.1.3 Componentes do zabbix

Conforme HORST; PIRES; DÉO (2015, p. 21), o Zabbix é composto de vários módulos, compreendendo o modulo Zabbix Server, Zabbix Agent, Zabbix Proxy, Java Gateway, Interface Web e o Banco de Dados, e esta seção irá elencá-los.

O módulo Zabbix Server é o componente principal do sistema, capaz de verificar remotamente os serviços de rede (web e e-mail) utilizando a checagem de forma simples, bem como é o componente central para que os agentes enviem informações e estatísticas a cerca da disponibilidade e integridade do equipamento que está sendo monitorado. O módulo recebe as informações, as processa, exibe relatórios, envia alertas e realiza ações pré-configuradas.

O módulo Zabbix Agent é a aplicação encarregada de coletar as informações dos dispositivos gerenciados e enviar ao Zabbix Server ou Zabbix Proxy. Também é capaz de acompanhar efetivamente o uso dos recursos e aplicações nos *hosts* gerenciados, tais como: processos, serviços, aplicativos em execução, disco rígido, entre outros.

O módulo Zabbix Proxy é semelhante ao Zabbix Server, podendo coletar dados de desempenho e disponibilidade de equipamentos gerenciados, porém com a responsabilidade de repassar as informações ao Zabbix Server. O Zabbix Proxy é capaz de coletar milhares de informações por segundo e armazená-la em seu banco de dados diminuindo assim a carga de processamento do Zabbix Server.

Interface Web (*Frontend*) é onde o administrador de rede mais interage com o Zabbix Server, possibilitando a customização de seu monitoramento. Nesse componente os usuários do Zabbix possuem acesso a mapas, gráficos e telas conforme regras de permissionamento.

O Banco de Dados (BD) é o componente responsável por armazenar os dados coletados pelo Zabbix Server ou Zabbix Proxy, bem como as configurações do sistema. O BD pode ser acessado tanto pelo Zabbix Server quanto pela Interface Web.

O Java Gateway a partir da versão 2.0, foi implementado como suporte nativo ao monitoramento das aplicações JMX (*Java Management Extension*) por meio do *daemon Zabbix Java Gateway*. É esse o processo responsável por recuperar os contadores do JMX.

#### 6.1.4 Arquitetura do Zabbix

Conforme demonstrado no item 6.1.3, a ideia é que cada componente seja instalado em um servidor/equipamento separado conforme a Figura 6.1, neste contexto o hardware de cada equipamento é dedicado à sua função, como por exemplo o Banco de Dados, onde é armazenado todas as coletas feitas pelo Zabbix Server. Isso requer um espaço em HD (*Hard Disk* – Disco Rígido) maior ou *Storage*, porém, em contrapartida a Interface Web não necessita de muito espaço em disco pois toda a base de informação estará armazenada no banco de dados de outro servidor.

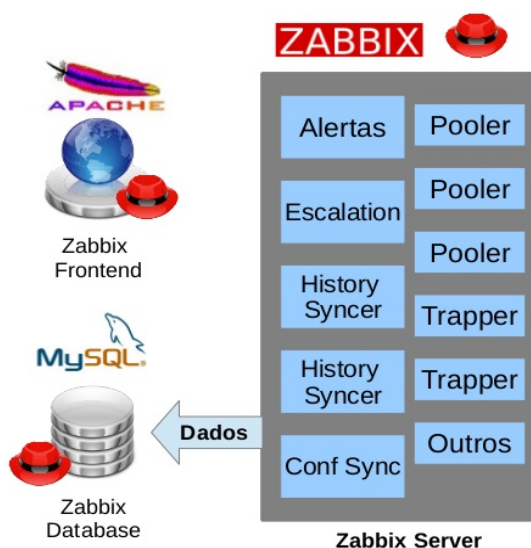


Figura 6.1: Principais componentes do Zabbix trabalhando de forma separada.

Fonte: <http://pt.slideshare.net/AlessandroSilva/monitoramento-enterprise-com-zabbixrhel>

Porém, conforme HORST; PIRES; DÉO (2015, p. 22), nada impede que se execute os componentes Zabbix Server, Interface Web e Banco de Dados em um único servidor, como pode ser visto na figura 6.2. Apesar desse formato de instalação ser comumente utilizado, o autor ressalta que pode haver várias razões para separá-los.

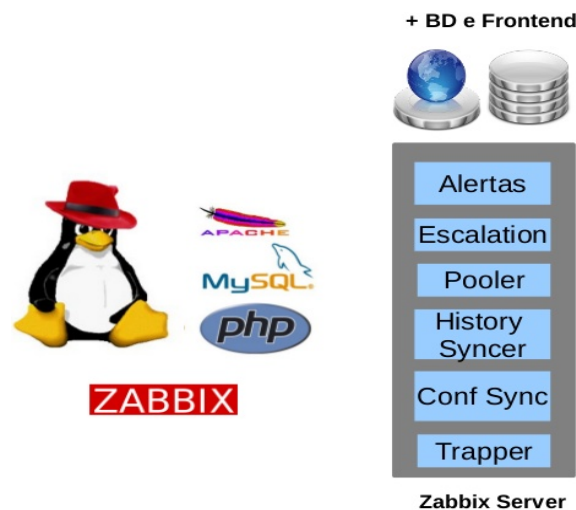


Figura 6.2: Zabbix Server unificado com o Banco de Dados e a Interface Web

Fonte: <http://pt.slideshare.net/AlessandroSilva/monitoramento-enterprise-com-zabbixrhel>

Muitas vezes é necessário o monitoramento de outras redes separadas por um *firewall* e centralizar as informações destas no Zabbix Server. Nesse contexto é incorporado outro componente importante para as coletas dos dados, o Zabbix Proxy. Este recebe os dados dos equipamentos monitorados em outra rede, armazena em seu Banco de Dados, e de forma ativa ou passiva conforme a Tabela 1, transfere os dados para o Zabbix Server HORST; PIRES; DÉO (2015, p. 22). Esse cenário pode ser visto na Figura 6.3.

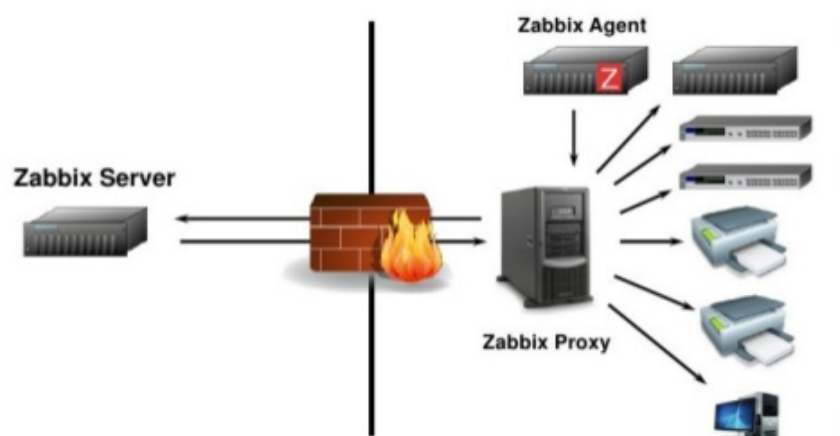


Figura 6.3: Zabbix Proxy recebe os dados e envia ao Zabbix Server

Fonte: <http://pt.slideshare.net/AlessandroSilva/monitoramento-enterprise-com-zabbixrhel>

### 6.1.5 Comunicação entre o Agente e o Zabbix Server

Existem 5 formas de comunicação entre o Agente gerenciado e o Zabbix Server, como podemos verificar na Tabela 1. Essa comunicação entre cliente e servidor varia de acordo com o tipo de item a ser coletado do dispositivo gerenciado.

*Tabela 1: Comunicação entre agente gerenciado e Zabbix Server*

Dispositivo Gerenciado	Porta / Protocolo	Descrição
Zabbix Agent Passivo	10050 / TCP	Neste modo de gerenciamento o Zabbix Server se conectar ao dispositivo gerenciado e solicita os itens que o usuário configurou para coletar, o agente Zabbix sempre aguarda a solicitação do Zabbix Server. (HORST; PIRES; DÉO, 2015, p. 24)
Zabbix Agent Ativo	10051 / TCP	O dispositivo gerenciado neste modo de gerenciamento, se conecta ao Zabbix Server, recebe a lista dos itens a serem monitorados, coleta os dados conforme o cronograma passado pelo Zabbix Server e periodicamente o envia. (HORST; PIRES; DÉO, 2015, p. 24)
JMX	10052 / TCP	É usado nos casos de monitoramento de servidores de aplicação Java, por meio do componente Java Gateway. (HORST; PIRES; DÉO, 2015, p. 24)
SNMP	161 / UDP	Neste modo de gerenciamento os dados coletados pelo Zabbix Server são aqueles que os fabricantes de equipamento implementaram conforme as normas RFC's.
IPMI	623 / UDP	O gerenciamento utilizado este modo é destinado ao monitoramento de recurso de hardware, ou seja, não depende do sistema operacional ter sido iniciado. Podemos monitorar temperaturas, voltagem, velocidade de HD. (LIMA, 2012).

Conforme apresentado na Tabela 1, para cada tipo de comunicação entre Dispositivo Gerenciado e Zabbix Server há uma porta padrão, porém nada impede que essa porta de comunicação seja alterada. É possível que um mesmo dispositivo seja monitorado pelo Zabbix Server por mais de uma forma, ou seja, monitoramento combinatório de acordo com o item a ser coletado. Como por exemplo, unir o monitoramento do Zabbix Agent Ativo com o monitoramento via protocolo IPMI. (HORST; PIRES; DÉO, 2015, p. 24)

## **7 IMPLANTAÇÃO DO ZABBIX NO IFSP CÂMPUS CAPIVARI**

O IFSP (Instituto Federal de São Paulo) é uma autarquia federal de ensino, técnico e tecnológico, fundada em 1909 como Escola de Aprendiz Artífices. Desde então receberam outros nomes como, Escola Técnica Federal de São Paulo e Centro Federal de Educação Tecnológica de São Paulo. Em 2008 houve a transição para Instituto Federal de Educação, Ciência e Tecnologia de São Paulo. Atualmente a instituição conta com 31 Câmpus Pleno e 2 Câmpus Avançado (IFSP, 2015).

O Câmpus Capivari foi criado em um ato oficial realizado em Brasília em fevereiro de 2010. Na ocasião, o Câmpus nasceu com a denominação de Câmpus Avançado Capivari e foi vinculado ao Câmpus Salto, porém suas atividades só tiveram início no dia 26 de Julho de 2010 com oferta dos cursos técnicos concomitantes ou subsequentes em Manutenção e Suporte em Informática e Química (IFSP CÂMPUS CAPIVARI, 2015).

Em Maio de 2013 o Instituto Federal de Capivari deixou de ser um Câmpus avançado e se tornou um Câmpus pleno, isso significa que conquistou independência em vários sentidos, como por exemplo, em processos de aquisições de bens patrimoniais, aumento significativo na verba para Câmpus, aumento o quadro de recursos humanos tanto docentes como técnicos administrativos (IFSP CÂMPUS CAPIVARI, 2015).

O Câmpus Capivari atua em 2 (duas) áreas de ensino: Informática e Química na formação técnica e superior, com os cursos técnico integrado e concomitante, tecnólogo em Análise e Desenvolvimento de Sistemas, tecnólogo em Processos Químicos e Licenciatura em Química. Também é ofertado, na modalidade de ensino a distância (EAD), o curso técnico em Administração (IFSP CÂMPUS CAPIVARI, 2015).

Atualmente o Câmpus Capivari oferece 240 vagas anuais e conta com 366 discentes<sup>1</sup> matriculados modalidade presencial, tanto nos cursos técnicos quanto nos cursos superior e na modalidade EAD o Câmpus oferta 50 vagas anualmente e conta com 80 discentes<sup>2</sup> no curso de Administração (IFSP CÂMPUS CAPIVARI, 2015).

---

1 CRE (Coordenadoria de Registros Escolares) do Câmpus Capivari – coletado em 19 de Out. 2015

2 Coordenadoria do Polo Capivari – Coordenador Rafael Wendel Pinheiro – coletado em 20 de Out. 2015

## **7.1 Cenário de pesquisa**

O estudo desse trabalho é sobre o monitoramento e gerenciamento de rede de computadores utilizando o Zabbix sendo que, para obter contribuições sobre a problemática citada no início deste trabalho, além do estudo bibliográfico, faz se necessário um estudo prático sobre a ferramenta. Para tal, foi feita a sua implantação nas redes administrativa, laboratório e Wi-Fi do Câmpus Capivari, com o objetivo de obter métricas de comparação em relação aos benefícios e características da implantação dessa ferramenta.

Para organização deste estudo, foi feito um levantamento junto a equipe de suporte de T.I. do Câmpus Capivari, a fim de identificar quais seriam os principais problemas existentes no Câmpus, os quais poderiam ser mitigados através de uma ferramenta de gerenciamento de rede. Este levantamento prévio apontou três problemas principais: (1) gerenciamento de impressoras, (2) gerenciamento de espaço em disco dos servidores e (3) indisponibilidade do Wi-Fi (HotSpot) do câmpus.

Os três problemas apresentados serão melhores especificados ao longo desta sessão, já que orientaram todo o trabalho prático.

### **7.1.1 Equipe de Suporte a T.I**

O Câmpus Capivari conta com uma equipe de 5 (cinco) profissionais sendo 2 (dois) Técnicos de Tecnologia da Informação e 3 (três) Técnicos de Laboratório de Informática ambos trabalham em regime de 6 hrs por dia de segunda a sexta-feira. Essa equipe é altamente qualificada, com 3 técnicos cursando a graduação, 1 cursando pós-graduação e 1 especialista.

O setor de T.I. está diretamente ligado a direção-geral do Câmpus por se tratar de setor estratégico de suporte a todas as áreas e fornece suporte nas mais variadas atividades de administração para os técnicos administrativos e atividades de ensino, pesquisa e extensão aos docentes e discentes. Para tal, o setor funciona com atendimento ininterrupto de 12 horas diárias, compreendidas entre 8h00 e 20h00.

### **7.1.2 Infraestrutura de T.I**

Em 2010 o Câmpus Capivari deu início as suas atividades acadêmicas e a infraestrutura de rede que existia foi projetada pela empreiteira que construiu o prédio. Infelizmente não há documentos a respeito desse projeto para expor nesse trabalho.

Em junho de 2013 a equipe de T.I. deu início a seus trabalhos e encontrou um cenário com algumas limitações, tanto no âmbito documental quanto nos equipamentos e infraestrutura de rede, conforme Tabela 2.

Tabela 2: Equipamentos existentes em junho de 2013

Fonte: Coordenadoria de Tecnologia da Informação do câmpus Capivari

Equipamentos do câmpus Capivari em 2013	
Rede Única	
Equipamento	Quantidade
<i>Access Point</i>	4
Computador ( <i>Desktop</i> )	140 <sup>3</sup>
Computador (Notebook)	3
Impressora de rede	1
Servidor Físico	4
<i>Switch</i>	7
<i>Switch</i> Gerenciável	1
Total de Equipamentos	160

Em nível documental, não temos informações sobre a responsabilidade pela elaboração e guarda desse projeto. Os equipamentos estavam desatualizados em nível de hardware e software, quanto a infraestrutura de rede, todo o Câmpus estava interligado por uma única rede física, ou seja, os discentes utilizavam a mesma rede que os funcionários do Câmpus, incorrendo em risco para a integridade e segurança dos dados.

A topologia de rede existente no Câmpus era do tipo estrela, onde todos os computadores se interconectavam por segmentos ponto a ponto. Toda a estrutura de cabeamento era do tipo par trançado, categoria CAT5e. A conexão de internet do Câmpus era fornecida pela RNP<sup>4</sup> e pela Prefeitura de Capivari, cujas velocidades de enlace eram de 4Mbps e 2Mbps respectivamente.

Atualmente a instituição conta com os equipamentos mais atualizados, com poder de processamento elevado e com garantia mínima de 1 ano, listados na Tabela 3.

Desde julho de 2013 foi planejada a reestruturação da rede, de forma que houve a separação física em 3 (três) redes. A primeira rede é dedicada somente para o uso administrativo e de docentes, onde todos os sistemas para atender as necessidades

---

<sup>3</sup> Valor aproximado.

<sup>4</sup> Rede Nacional de Ensino e Pesquisa

administrativas e acadêmicas estão em funcionamento. Esta rede está estruturada com cabeamento CAT6. A segunda rede é destinada ao uso acadêmico, ou seja, para fins didáticos entre docentes e discentes, todo o cabeamento ainda é antigo par trançado CAT5e. A terceira rede é destinada tanto para o público interno quanto externo (comunidade em geral/visitantes), o cabeamento até o access point está estruturado com CAT6.

Os acessos em todas as redes são realizados através de autenticação do usuário previamente cadastrado junto ao departamento de TI do câmpus, o servidor de centralização das contas de usuário é o OpenLdap.

*Tabela 3: Equipamentos existentes no Câmpus Capivari em Junho de 2015*

<b>Equipamentos do Câmpus Capivari em 2015</b>			
	<b>Rede Administrativa</b>	<b>Rede Laboratorial</b>	<b>Rede Wi-Fi (HotSpot)</b>
<b>Equipamento</b>	<b>Quantidade</b>	<b>Quantidade</b>	<b>Quantidade</b>
Access Point	1	1	2
Computador ( <i>Desktop</i> )	90 <sup>5</sup>	170 <sup>6</sup>	-
Computador ( <i>Notebook</i> )	5	26	-
DVR	3	-	-
Impressora de Rede	4	-	-
Pabx	1	-	-
Servidor Físico	5	2	-
Servidor Virtual	23	18	2
Switch	-	2	-
Switch Gerenciável	11	6	-
Total de Equipamento	143	225	4

Os servidores que atende a demanda tanto administrativa quanto acadêmica são equipamento adquiridos em dezembro de 2014, cujo as especificações estão na Tabela 4.

<sup>5</sup> Com base no número de servidores divulgados no site institucional acessado em 1 de novembro de 2015  
<http://www.ifspcapivari.com.br/corpo-administrativo/>  
<http://www.ifspcapivari.com.br/corpo-docente/>

<sup>6</sup> Valor aproximado.



Tabela 4: Servidores Físicos que atende as redes administrativa e laboratório

Servidores Físicos	
Dell PowerEdge R720	<ul style="list-style-type: none"> <li>• Processador Intel Xeon E5-2650 8 core, 2.2 Ghz</li> <li>• 256 GB Memória RAM</li> <li>• 6 HD's SAS 600GB, 15k RPM, Hot Plug</li> </ul>
HP ProDesk 600 (Firewall)	<ul style="list-style-type: none"> <li>• Processador Intel Core i5, 3.2 Ghz</li> <li>• 4 GB Memória RAM</li> <li>• 1 HD 1 Tera Byte</li> </ul>

O cenário descrito pode ser visualizado na Figura 7.1, onde pode-se verificar as três redes separadas. Por questão de privacidade, segurança e integridade da infraestrutura de rede, o departamento de T.I não divulgou os IP's dos *hosts*, contudo é claramente possível entender o diagrama da rede.

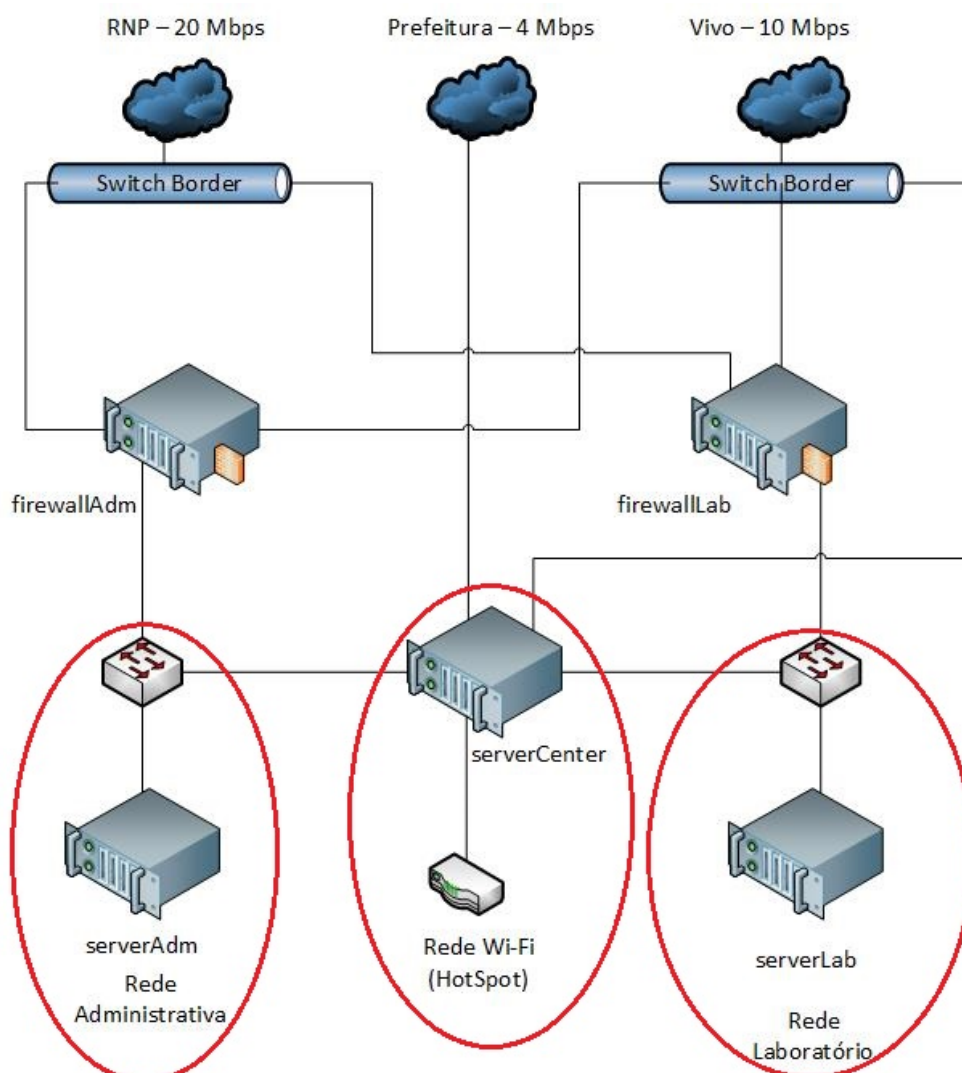


Figura 7.1: Estrutura de Rede do Câmpus Capivari

Fonte: Departamento de T.I do Câmpus Capivari

## 7.2 Implementação do Zabbix

Para ambiente de implementação do Zabbix, foi definido que os ativos a serem monitorados são recursos, serviços e aplicações com maiores incidências de reclamação dos usuários da instituição.

Impressoras de rede apontando os níveis de tóner e quantidade de cópias/impressões, monitoramento de HD analisando a quantidade total e o espaço livre para armazenamento de arquivos, Wi-Fi monitorando a disponibilidade do equipamento e da área de autenticação (Captive Portal) para navegação são exemplos de recursos a serem monitorados.

O Zabbix Server foi instalado através do código fonte em um servidor virtual, estrategicamente localizado no servidor físico (serverCenter), possibilitando assim a comunicação entre todos os *hosts* das 3 (três) redes de computadores do Câmpus Capivari. As configurações desse servidor virtual estão listadas na Tabela 5.

A instalação se deu de forma unificada entre Zabbix Server, Banco de Dados e Interface Web, em um mesmo servidor virtual. As configurações do Zabbix Server foram mantidas com o padrão de instalação. Todas essas configurações foram escolhidas com base na quantidade aproximada de *hosts* a serem monitorados e na documentação<sup>7</sup> oficial do Zabbix.

*Tabela 5: Especificação do servidor virtual com Zabbix Server*

Zabbix Server	
Sistema Operacional	GNU/Linux
Distribuição	CentOS 6.5 (RHEL) 64 bits
Memória RAM	4 GB
Disco Rígido	15 GB (SATA)
2 Processadores	Intel(R) Xeon(R) E5-2650 v2 2.60GHz
3 Placas de Rede	Intel PRO/1000 MT Desktop (Placa em modo bridge)

Para a instalação do Zabbix, foram baixados os arquivos-fonte do site oficial da Zabbix SIA<sup>8</sup> utilizamos a versão 2.4, pois atualmente é a versão mais estável. Foi realizado a criação dos binários (arquivos executáveis), para esses binários é necessário a existência do usuário **zabbix**.

<sup>7</sup> <https://www.zabbix.com/documentation/2.4/manual/installation/requirements>

<sup>8</sup> <https://www.zabbix.com/download.php>

Na instalação do Banco de Dados foi escolhido o PostgreSQL e utilizado a versão 9.3, o pré-requisito da Zabbix SIA é a versão 8.1 ou superior. Todas as configurações e criação de tabelas estão descritas no Apêndice A.

Para a Interface Web foi utilizado o servidor Apache versão 2.2.15 com interpretador PHP versão 5.3.3, o pré-requisito da Zabbix SIA é a versão 1.3.12 ou superior. Todas as do PHP estão descritas no Apêndice A.

O arquivo de configuração do Zabbix Server é bem abrangente e possui vários parâmetros, conforme Zabbix SIA<sup>9</sup>, porém, o cenário do câmpus foi configurado apenas o básico para o funcionamento do Zabbix Server, conforme descrito no Apêndice A.

Após a configuração do servidor com os serviços necessários para o funcionamento do Zabbix Server partiu-se para as configurações no navegador e a inicialização de monitoramentos. O primeiro *host* a ser monitorado foi o próprio Zabbix Server com o *template* padrão.

Os *templates* são modelos de regras de coleta, alerta e representação gráfica que podem facilmente se aplicado a elementos monitorados. Várias ferramentas de monitoramento suportam essa funcionalidade, entretanto o Zabbix é uma das poucas ferramentas que suporta esse recurso em conjunto com heranças de propriedades.

Uma analogia seria como a forma de bolo, que pode ser usado para fazer bolos de vários sabores como um mesmo formato. Outra analogia aos profissionais de desenvolvimento (programadores), é uma Classe que define as características de um objeto em POO. Esse recurso traz muita agilidade ao administrador caso ele tenha um parque tecnológico uniforme, ou seja, com equipamentos e serviços padrões em seu ambiente de monitoramento.

## 7.3 Problema 1: Gerenciamento de Impressoras

### 7.3.1 Pré implementação

O Câmpus Capivari possui em sua estrutura atual, impressoras de rede que estão em operação conforme demonstrado na Tabela 3. Dentre as impressoras de rede, a instituição possui 2 (dois) modelos diferentes, que são: (1) HP Color LaserJet Enterprise M750dn, (2) Sharp MX-M314N.

O modelo 1 faz parte do *hall* de patrimônios do câmpus adquirido em outubro de

---

9 [https://www.zabbix.com/documentation/2.4/manual/appendix/config/zabbix\\_server](https://www.zabbix.com/documentation/2.4/manual/appendix/config/zabbix_server)

2014<sup>10</sup> através de processo licitatório, foram compradas 3 unidades e o câmpus fez permuta com o câmpus Boituva e manteve somente 1 unidade em sua rede administrativa.

O modelo 2 foi adquirido em março de 2015<sup>11</sup> através de um contrato *outsourcing* válido por 1 ano, nesse contrato contempla 3 equipamentos de impressora, 15.000 (mil) cópias/impressões mensais, tônner para cada impressora e um tônner reserva. Os fiscais de contrato precisam acompanhar diariamente esses números de impressões em cada dispositivo e expor esses dados a todos os funcionários do câmpus, esse processo se dá de forma manual e controlado por planilha eletrônica, conforme demonstrado na Figura 7.2.

Quantidade de Impressão - Definições de Sistema - Contagem Total - Preto e Branco							
Data do Movimento	Data Leitura Contador	Impressora CAD 5624	Impressora Corredor 5623	Impressora CRE 5622	TOTAL	Acumulado	Total/Dia
31/08/2015	01/09/2015	32521	45268	6402	84191	84191	793
01/09/2015	02/09/2015	32839	46142	6475	85456	1265	1265
02/09/2015	03/09/2015	33143	46630	6543	86316	2125	860
03/09/2015	04/09/2015	33348	47290	6558	87196	3005	880
04/09/2015	08/09/2015	33755	47604	6597	87956	3765	760
08/09/2015	09/09/2015	34057	48076	6676	88809	4618	853
09/09/2015	10/09/2015	34280	48802	6740	89822	5631	1013
10/09/2015	11/09/2015	34399	49392	6804	90595	6404	773
11/09/2015	14/09/2015	34628	49924	7009	91561	7370	966
14/09/2015	15/09/2015	34777	50768	7176	92721	8530	1160
15/09/2015	16/09/2015	34977	51409	7203	93589	9398	868
16/09/2015	17/09/2015	35157	51568	7267	93992	9801	403
17/09/2015	18/09/2015	35817	51568	7399	94784	10593	792
18/09/2015	21/09/2015	36477	51568	7531	95576	11385	792
21/09/2015	22/09/2015	36672	52033	7563	96268	12077	692
22/09/2015	23/09/2015	36931	52682	7708	97321	13130	1053
23/09/2015	24/09/2015	37063	53591	7837	98491	14300	1170
24/09/2015	25/09/2015	37231	53893	7913	99037	14846	546
25/09/2015	28/09/2015	37312	54118	7979	99409	15218	372
28/09/2015	29/09/2015	37477	54734	8020	100231	16040	822
29/09/2015	30/09/2015						
30/09/2015	01/10/2015						

Figura 7.2: Controle de cópias/impressões do Câmpus Capivari

Fonte: Fiscalização de Contrato de Reprografia do Câmpus Capivari

Percebe-se pela Figura 7.2 que no dia 29/09/2015, na coluna Acumulado o limite de cópias/impressões do mês excedeu em 1.040 páginas e esse tipo de ocorrência gera transtornos muito grande para a instituição, pois o contrato permite apenas 15.000 páginas.

Outro problema relatado pela fiscalização desse contrato são os níveis de tônner de cada impressora. Por 2 vezes no ano de 2015 aconteceram incidentes relacionados ao tônner, em

10 Dado obtido na Coordenadoria de Almoxarifado e Patrimônio do Câmpus Capivari, em 26/10/2015.

11 Dado obtido junto a Fiscalização de Contratos do Câmpus Capivari, em 26/10/2015

uma das ocasiões o equipamento ficou inoperante por 2 dias consecutivos.

### 7.3.2 Processo de Implementação do Zabbix

Pensando nisso, foi realizado algumas pesquisas relacionadas a implementação do Zabbix para mitigar essas ocorrências. Visto que o Zabbix trabalha com *templates*, optou-se por implementar o monitoramento das 3 (três) impressoras utilizando-o um *template* com o nome de **Template Sharp**.

Para executar a coleta nesse tipo de *host* é necessário utilizar o protocolo SNMP, visto que não é possível instalar um Zabbix Agent no equipamento. O protocolo SNMP foi habilitado na versão 2. Também se fez necessário um estudo profundo relacionado a MIB do equipamento desse fabricante, em seguida foi possível identificar quais OID's correspondem aos itens que desejamos coletar.

Nesse *template* foram criados 2 grupos (aplicações) com os nomes de **Contador** e **tôner**, já que essas aplicações são importantes para organizar e agrupar os itens. Na aplicação Contador foram criados 2 itens relevantes que são: **quantidade de cópia/impressão** e **somaImpressao**. Na aplicação tôner, foi criado apenas 1 item relevante que é: **nível atual de tôner**.

As impressoras foram criadas manualmente no Zabbix Server e associadas ao Template Sharp. Com isso as configurações realizadas no Template Sharp são herdadas nos *hosts* associados. Também é possível criar as impressoras de forma automatizada, porém não é objeto de estudo deste tópico.

Na aplicação Contador, para coletar a quantidade de cópia/impressão foi definido que esse item é do tipo Agente SNMPv2 e um nome da chave *QtyCopyPrint*. No SNMP OID que é a identificação encontrada na MIB da impressora relacionada a quantidade de cópias e impressões e o SNMP COMMUNITY foi colocado como *public*, essas coletas são realizadas a cada 10 minutos. É possível visualizar as configurações na Figura 7.3.

**ZABBIX** Ajuda | Suporte | Imprimir | Configurações | Desconectar

Monitoramento | Inventário | Relatórios | **Configuração** | Administração | Zabbix Server

Grupos de hosts | Templates | **Hosts** | Manutenção | Ações | Telas | Slideshows | Mapas | Autobusca | Serviços de TI | Pesquisar

Histórico: Dados personalizados do usuário » Configuração da autenticação » Dashboard » Configuração dos hosts » Configuração dos templates

**CONFIGURAÇÃO DE ITENS** Criar item

**Itens**

Mostrando 1 para 3 de 3 encontrados

Exibir filtro

« Lista de templates **Template: Template Sharp** Aplicações (2) Itens (3) Triggers (1) Gráficos (1) Telas (0) Regras de descoberta (0) Cenários web (0)

Assistente	Nome	Triggers	Chave	Intervalo	Histórico	Estatísticas	Tipo	Aplicação
<input type="checkbox"/>	Nível Atual de Toner		LevelToner	300	90	365	Agente SNMPv2	Toner
<input type="checkbox"/>	Quantidade de Cópias/Impressões	Triggers (1)	QtdCopyPrint	30	90	365	Agente SNMPv2	Contador
<input type="checkbox"/>	SomaImpressao		grpsum["500_Impressoras","QtdCopyPrint",last,0]	1	90	365	Zabbix agregado	Contador

Ativar selecionadas Ir (0)

Zabbix 2.4.6 é uma marca registrada 2001-2015 pela Zabbix SIA Conectado como 'Admin'

Figura 7.3: Itens relevantes para o monitoramento de impressoras Sharp

Para ter a quantidade total de páginas impressas em todas as impressoras foi criado o item quantidade total, esse item é do tipo calculado pois tem a função de efetuar cálculos com itens coletados. A chave selecionada foi a mesma criada anteriormente *QtdCopyPrint* e no campo da fórmula foi implementado o cálculo simples do item somando cada impressora. O tempo de cálculo foi definido em 15 minutos para atualizações da quantidade de páginas impressas.

Foram criadas 3 *Triggers*<sup>12</sup> (Gatilho) para alertar os fiscais de contrato sobre a quantidade de páginas impressas em determinado período de tempo.

Uma *trigger* é disparada quando a quantidade alcançou 13.000 páginas e o nível de severidade configurado para essa quantidade de páginas foi tratado como **Média**, para lembrar que já está próximo ao limite daquele mês. Outra *trigger* é disparada no momento em que o número de páginas impressas alcançou o nível de 14.000 e o nível de severidade neste caso foi atribuído com **Alto**, pois já houve ocasião em que foram impressas mais de 1.200 páginas em um mesmo dia. A terceira *trigger* é executada quando o limite de impressões já excedeu o valor máximo de 15.000 páginas no mês, neste caso o nível de severidade escolhido foi de **Desastre**.

Pensando na facilidade visual desses dados foi elaborado um gráfico que demonstra a quantidade de páginas impressas em cada impressora e a *trigger* com o limite de impressão do mês.

<sup>12</sup> Triggers são expressões lógicas que “avaliam” os dados reconhecidos por itens e representam o estado atual do sistema. Fonte: [https://www.zabbix.com/documentation/2.4/manual/config/triggers?s\[\]=trigger](https://www.zabbix.com/documentation/2.4/manual/config/triggers?s[]=trigger)

Na aplicação *tôner*, para coletar o nível de *tôner* em cada impressora assim como a aplicação *Contador*, foi marcado que esse item é do tipo *Agente SNMPv2* e um nome da chave *LevelToner* (nome a sua escolha), no *SNMP OID* é a identificação encontrada na *MIB* da impressora relacionada ao nível de *tôner* e o *SNMP COMMUNITY* colocamos como *public*, essas coletas são realizadas a cada 12 horas. É possível visualizar as configurações na Figura 7.4.

The screenshot shows the Zabbix web interface for configuring a new item. The top navigation bar includes links for 'Ajuda', 'Suporte', 'Imprimir', 'Configurações', and 'Desconectar'. Below this, there are tabs for 'Monitoramento', 'Inventário', 'Relatórios', 'Configuração', and 'Administração'. The 'Configuração' tab is active, and the 'Configuração de itens' sub-tab is selected. The breadcrumb trail shows the path: 'Histórico: Configuração de gráficos » Gráficos personalizados » Configuração de itens » Gráficos personalizados » Configuração de itens'. The main heading is 'CONFIGURAÇÃO DE ITENS'. Below this, there are links for 'Lista de templates', 'Template: Template Sharp', 'Aplicações (2)', 'Itens (5)', 'Triggers (1)', 'Gráficos (2)', 'Telas (0)', 'Regras de descoberta (0)', and 'Cenários web (0)'. The 'Item' configuration form is displayed with the following fields: 'Nome' (Nível Atual), 'Tipo' (Agente SNMPv2), 'Chave' (LevelToner), 'SNMP OID' (1.3.6.1.2.1.43.11.1.1.9.1.1), 'Comunidade SNMP' (public), 'Porta' (161), 'Tipo de informação' (Numérico (inteiro sem sinal)), 'Tipo de dados' (Decimal), 'Unidades' (%), 'Usar multiplicador customizado' (checkbox), 'Intervalo atualização (em seg)' (43200), and 'Intervalos flexíveis' (table with columns Intervalo, Período, Ação). The 'Intervalos flexíveis' table is empty, and a 'Novo intervalo flexível' button is visible. Below the table, there are fields for 'Período de retenção do histórico (em dias)' (90), 'Período de retenção das estatísticas (em dias)' (365), 'Armazenar valor' (Sem alterar), 'Mostrar valor' (Sem alterar), and 'Nova aplicação' (Aplicações). The 'Aplicações' dropdown menu is open, showing options: '-Nenhum-', 'Contador', and 'Toner'.

Figura 7.4: Configuração para coletar o nível atual de *tôner* da impressora

Infelizmente o fabricante dessa impressora disponibilizou em sua *MIB* apenas 5 valores relacionados ao nível de *tôner*, são eles: 0% quando o *tôner* já acabou; 25%, 50% e 75% quando existe *tôner*; e 100% quando o nível do *tôner* está completo.

Foi criado 2 *triggers* que são acionadas nos valores mais baixos relacionado ao nível de *tôner*. A primeira *trigger* é disparada quando o nível de *tôner* chega em 25%, com isso há tempo hábil para o fiscal de contrato conferir se possui em estoque *tôner* reserva, o grau de severidade escolhido para esse tipo de alerta é **Alto**. Já na segunda *trigger* o evento é alertado quando o nível de *tôner* chegou em 0% ou seja, acabou o *tôner*, o grau de severidade

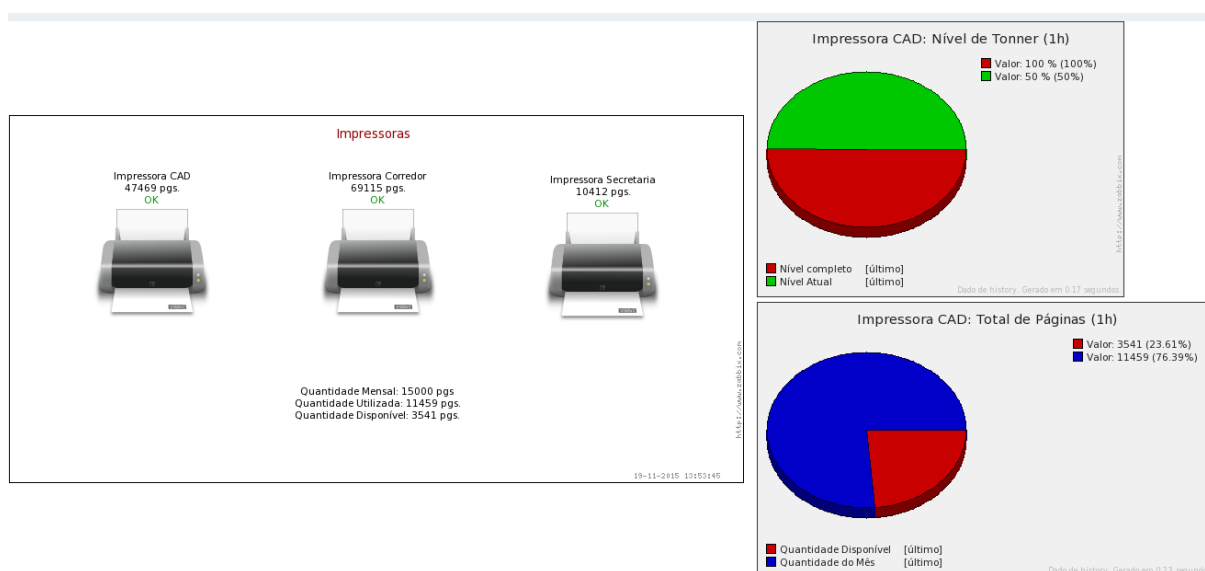


escalonado para esse tipo de evento foi definido como **Desastre**.

Seguindo o padrão da aplicação anterior, para visualizar com maior clareza o nível de t  ner, foi elaborado um gr  fico em forma de pizza, isso deixa os fiscais de contrato com maior clareza do n  vel de t  ner em tempo real.

### 7.3.3 Benef  cios p  s implementa  o

Ap  s a implementa  o do monitoramento das p  ginas impressas das impressoras, percebe-se que o uso da planilha para anota  es de c  pias/impress  es foi abolido pelos fiscais de contrato, pois a quantidade de erros que ocorriam no momento da digita  o era grande e o tempo que era despendido para realizar essas coletas di  rias era muito alto.



*Figura 7.5: Impressoras monitoradas - Quantidade de impress  o e n  vel de t  ner*

Conforme figura 7.5, percebe-se que os fiscais de contrato tem uma vis  o macro da quantidade de impress  o e n  vel de t  ner de todas as impressoras monitoradas, podendo assim realizar o monitoramento fiel de cada equipamento.

Outro benef  cio que podemos elencar    relacionado a tempo para coletar as informa  es manualmente, logo esse tempo est   diretamente ligado as finan  as da empresa.

O tempo despendido para realizar essas coletas e inseri-las na planilha eletr  nica, gira em torno de 5 minutos por dia. O valor da hora trabalhada de um t  cnico    de aproximadamente R\$ 18,00. Ou seja, em 1 ano s  o gastos 20 horas de m  o de obra para realizar essa tarefa e financeiramente R\$ 360,00 pagos pela institui  o.



Quanto ao controle de t  ner que n  o existia, teve um grande ganho relacionado ao *downtime* das impressoras devido a falta de t  ner, pois a equipe de fiscaliza  o do contrato sempre mant  m o equipamento operacional e quantidade suficiente de t  ner reserva.

Outra importante quest  o a ser citada   relacionada aos preju  zos n  o tang  veis em exceder o limite das impress  es ou a falta de t  ner em um per  odo de impress  es de provas e documentos importantes, visto que se trata de institui  o de ensino.

## **7.4 Problema 2: Monitoramento de espa  o em disco dos servidores**

### **7.4.1 Pr   implementa  o**

O C  mpus Capivari possui em sua estrutura atual 3 servidores f  sicos que atende a demanda tanto da  rea acad  mica quanto da  rea administrativa, conforme demonstrado na Figura 7.1. Nesses servidores f  sicos existem diversos servidores virtuais que disponibilizam os servi  os acad  micos e administrativos, exemplo servidor Samba que   respons  vel pela guarda e compartilhamento de arquivos, tanto para os discentes e docentes quanto para os t  cnicos administrativos utilizam desse recurso.

E para manter redund  ncia e backup desses servidores virtualizados o c  mpus possui 6 HD's externos destinados a esse fim. A equipe de suporte t  cnico do c  mpus desenvolveu um *script* para realizar o desligamento dos servidores virtuais para fins de *backup* das m  quinas virtuais. Ap  s a realiza  o da c  pia desses servidores no pr  prio servidor f  sico, ele liga todos os servidores virtuais exceto os *backups*, o servidor f  sico realiza a c  pia dos servidores que est  o desligados em HD's externos.

Na ocasi  o, os HD's externos possuem capacidade de espa  o limitado, em no m  ximo 2TB, com isso   necess  rio manter o controle quanto ao espa  o em disco utilizado em cada servidor virtual.

Em duas ocasi  es foram relatados por discentes que n  o estavam conseguindo guardar arquivos dentro do servidor Samba destinado ao uso acad  mico. A equipe de suporte do c  mpus identificou que o espa  o destinado a esse servidor j   n  o era suficiente, com isso houve a necessidade em aumentar o espa  o em disco do servidor Samba e por certo momento o servi  o ficou indispon  vel.

Para evitar esse tipo de inconveni  ncia novamente, realizamos a implementa  o do

monitoramento do espaço em disco de cada servidor virtualizado do câmpus, minimizando assim futuros problemas.

### 7.4.2 Processo de Implementação do Zabbix

Para realizar coletas das informações de todos os servidores virtuais tanto da rede administrativa quanto da rede acadêmica foi instalado o Zabbix Agent, visto que é altamente recomendado instalá-lo onde é possível, pois a gama de informações capturadas e tabuladas para o Zabbix Server é bem maior.

O agente instalado foi implementado no modo ativo, ou seja, periodicamente envia os dados que o Zabbix Server solicitou em sua primeira conexão. As configurações utilizadas nos agentes foram feitas conforme o padrão, ou seja, com os parâmetros mínimos para a troca de informações entre o agente e gerente, visto que neste momento não se faz necessário o uso de outros parâmetros. Conforme Figura 7.6.

```
# vi /etc/zabbix/zabbix_agentd.conf
```

**Conteúdo Mínimo do Arquivo:**

```
Server=IP do Servidor do Zabbix
Hostname=Nome da máquina Cliente
StartAgents=3
DebugLevel=3
PidFile=/var/tmp/zabbix_agentd.pid
LogFile=/tmp/zabbix_agentd.log
Timeout=3
```

Figura 7.6: Configuração mínima do Zabbix Agent

Fonte: <http://zabbixbrasil.org/wiki/tiki-index.php?page=Instalacao+de+Agente+Zabbix+em+ambiente+Linux>

Foi utilizado outro recurso muito importante incorporado a segunda versão do Zabbix que é a autobusca interna ou LLD (*Low Level Discovery*). Este recurso agrega a descoberta automatizada da rede com o que existe em cada equipamento, possibilitando assim criar regras de descoberta interna no equipamento, executá-las periodicamente e a partir dos dados descobertos pelo LLD o perfil de monitoração do *host* poderá ser modificado.

Para incidentes relativos ao problema em questão, foi criado um *template* cujo a função é padronizar aplicação, itens, *triggers*, gráficos e mapas em todos servidores monitorados, visto que a infraestrutura de servidores do câmpus Capivari segue um padrão de

configuração e sistema operacional para todas as redes. O nome utilizado para esse *template* é **Template OS Linux**.

Para utilizarmos a autobusca foi implementado uma regra de descoberta, cujas funcionalidades desta regra é descobrir quantas e quais partições existentes e qual o tamanho dessa partição.

Foram criados protótipos de itens, capaz de capturar o espaço total da partição, sua função de aferir qual o espaço de cada partição. Outro protótipo relevante é espaço utilizado no disco, que verifica a quantidade de espaço que está sendo utilizado. E o terceiro protótipo de item verifica o espaço livre dentro da partição.

**ZABBIX** Ajuda | Suporte | Imprimir | Configurações | Desconectar

Monitoramento | Inventário | Relatórios | **Configuração** | Administração | Zabbix Server

Grupos de hosts | Templates | **Hosts** | Manutenção | Ações | Telas | Slideshows | Mapas | Autobusca | Serviços de TI | Pesquisar

Histórico: Configuração dos templates » Configuração de itens » Dados recentes » Configuração dos templates » Configuração de regras de descoberta

**CONFIGURAÇÃO DE PROTÓTIPOS DE ITENS** Criar protótipo de item

**Protótipos de itens de Montado descoberta de sistema de arquivo**

Mostrando 1 para 5 de 5 encontrados

« Lista de templates **Template: Template OS Linux** « Lista de descoberta **Autobusca: Montado descoberta de sistema de arquivo** Protótipos de itens (5)

Protótipos de trigger (4) Protótipo de gráficos (1) Protótipos de host (0)

<input type="checkbox"/>	Nome	Chave	Intervalo	Histórico	Estatísticas	Tipo	Aplicações	Status
<input type="checkbox"/>	Espaco de disco utilizado em {#FSNAME}	vfs.fs.size[{#FSNAME},used]	60	7	365	Agente Zabbix	Sistema de Arquivo	Ativo
<input type="checkbox"/>	Espaco livre de disco em {#FSNAME}	vfs.fs.size[{#FSNAME},free]	60	7	365	Agente Zabbix	Sistema de Arquivo	Ativo
<input type="checkbox"/>	Espaco livre de disco em {#FSNAME} (porcentagem)	vfs.fs.size[{#FSNAME},pfree]	60	7	365	Agente Zabbix	Sistema de Arquivo	Ativo
<input type="checkbox"/>	Inodes livre em {#FSNAME} (percentage)	vfs.fs.inode[{#FSNAME},pfree]	60	7	365	Agente Zabbix	Sistema de Arquivo	Ativo
<input type="checkbox"/>	Total de espaco em disco no {#FSNAME}	vfs.fs.size[{#FSNAME},total]	3600	7	365	Agente Zabbix	Sistema de Arquivo	Ativo

Ativar selecionadas Ir (0)

Zabbix 2.4.6 é uma marca registrada 2001-2015 pela Zabbix SIA | Conectado como 'Admin'

Figura 7.7: Protótipo de item que captura o espaço de HD

Para alertar os técnicos de TI do câmpus, foi desenvolvido alguns protótipos *triggers* relacionados ao espaço de disco.

O primeiro alerta é disparado quando o espaço livre do disco está menor que 20 %, neste caso o grau de severidade foi definido como **Atenção**. Outra *trigger* é disparada no momento em que o espaço livre de disco atingiu 10% do disco e o grau de severidade escalonado para este alerta é considerado como **Alto**. O limite máximo para o disparo terceira *trigger* foi definido com 5% livre, neste caso o grau de severidade foi definido como **Desastre**. Essas *triggers* pode ser visualizada na Figura 7.7.

**ZABBIX** Ajuda | Suporte | Imprimir | Configurações | Desconectar

Monitoramento | Inventário | Relatórios | **Configuração** | Administração Zabbix Server

Grupos de hosts | Templates | **Hosts** | Manutenção | Ações | Telas | Slideshows | Mapas | Autobusca | Serviços de TI Pesquisar

Histórico: Configuração de itens » Dados recentes » Configuração dos templates » Configuração de regras de descoberta » Configuração dos protótipos de item

**CONFIGURAÇÃO DE PROTÓTIPOS DE TRIGGERS** Criar protótipo de trigger

**Protótipos de trigger de Montado descoberta de sistema de arquivo**

Mostrando 1 para 4 de 4 encontrados [ Esconder protótipos de trigger desativado ]

« Lista de templates **Template: Template OS Linux** « Lista de descoberta **Autobusca: Montado descoberta de sistema de arquivo** Protótipos de itens (5)

Protótipos de trigger (4) Protótipo de gráficos (1) Protótipos de host (0)

Severidade	Nome	Expressão	Status
Desastre	Espaco disponível em disco menor que 5% em {#FSNAME}	{Template OS Linux:vfs.fs.size({#FSNAME},pfree).last(0)}<5	Ativo
Alta	Espaco disponível em disco menor que 10% em {#FSNAME}	{Template OS Linux:vfs.fs.size({#FSNAME},pfree).last(0)}<10	Ativo
Atenção	Espaco disponível em disco menor que 20% em {#FSNAME}	{Template OS Linux:vfs.fs.size({#FSNAME},pfree).last(0)}<20	Ativo
Atenção	Inodes livres é inferior a 20% no volume de {#FSNAME}	{Template OS Linux:vfs.fs.inode({#FSNAME},pfree).last(0)}<20	Ativo

Ativar selecionadas Ir (0)

Zabbix 2.4.6 é uma marca registrada 2001-2015 pela Zabbix SIA Conectado como 'Admin'

Figura 7.8: Protótipo de triggers para espaço disponível em HD

Outra implementação realizada utilizando o LLD foi o protótipo de gráfico que é capaz de gerar gráficos do tipo torta para cada partição encontrada. Os itens escolhidos para apresentar nesse gráfico são, total de espaço em disco e o espaço livre no disco.

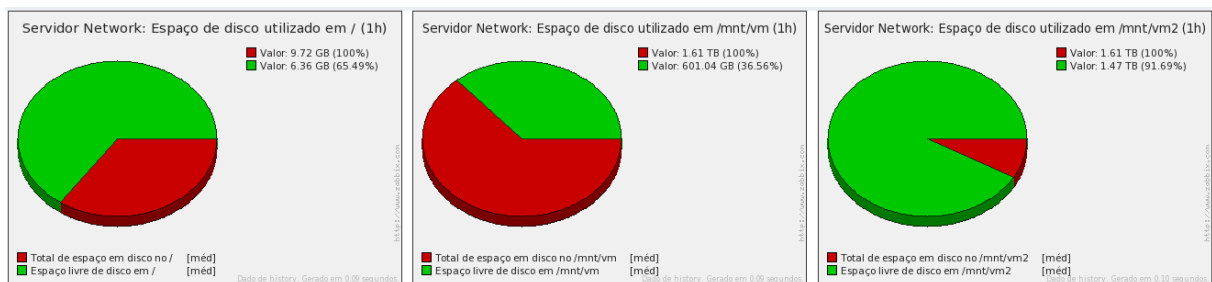


Figura 7.9: Gráficos das partições do servidor

Pensando na facilidade visual das partições de cada HD em todos os servidores monitorados, foi implementado uma tela de visualização que o usuário possa acompanhar qualquer servidor virtual.

## Benefícios pós implementação

Com essa implementação podemos perceber que as chances da superlotação do espaço de cada partição podem ser minimizadas, visto que os alertas sonoros e visuais antecipam os técnicos de TI do câmpus, ou seja, manutenção preventiva. Na Figura 7.8 temos o gráfico de um dos servidores com suas respectivas partições, nesse caso o espaço em verde corresponde ao espaço livre do disco.

Conforme relato houve caso do espaço do HD chegou ao limite sem o conhecimento da equipe de TI. O Zabbix veio a agregar esse benefício de manter a equipe de TI do câmpus sempre informado quanto ao real espaço utilizado do HD visualizando através dos gráficos ou alertando-os através das *triggers*.

## **7.5 Problema 3: Indisponibilidade do Wi-Fi (HotSpot) do câmpus**

### **7.5.1 Pré implementação**

A Coordenadoria de Tecnologia da Informação do câmpus Capivari disponibiliza uma rede Wi-Fi para o uso dos dispositivos móveis dos discentes, docentes, técnicos administrativos e comunidade externa.

Nesta rede sem fio é utilizado um servidor virtual com 2 placas de rede, cuja placa é ligada a LAN e a outra ligada a WAN onde recebemos link fornecido pela Prefeitura de Capivari.

Neste servidor virtual foi implementado o sistema operacional FreeBSD com o software PfSense versão 2.2.4-Release este software possui diversos serviços executando para o funcionamento do captive portal. Dentre eles o DHCP Service que distribui ip aos *hosts* conectados, o FreeRADIUS Service que se conecta a base de usuários externa, o squid versão 4.3.4 que faz a filtragem de conteúdo entre o usuário e a internet.

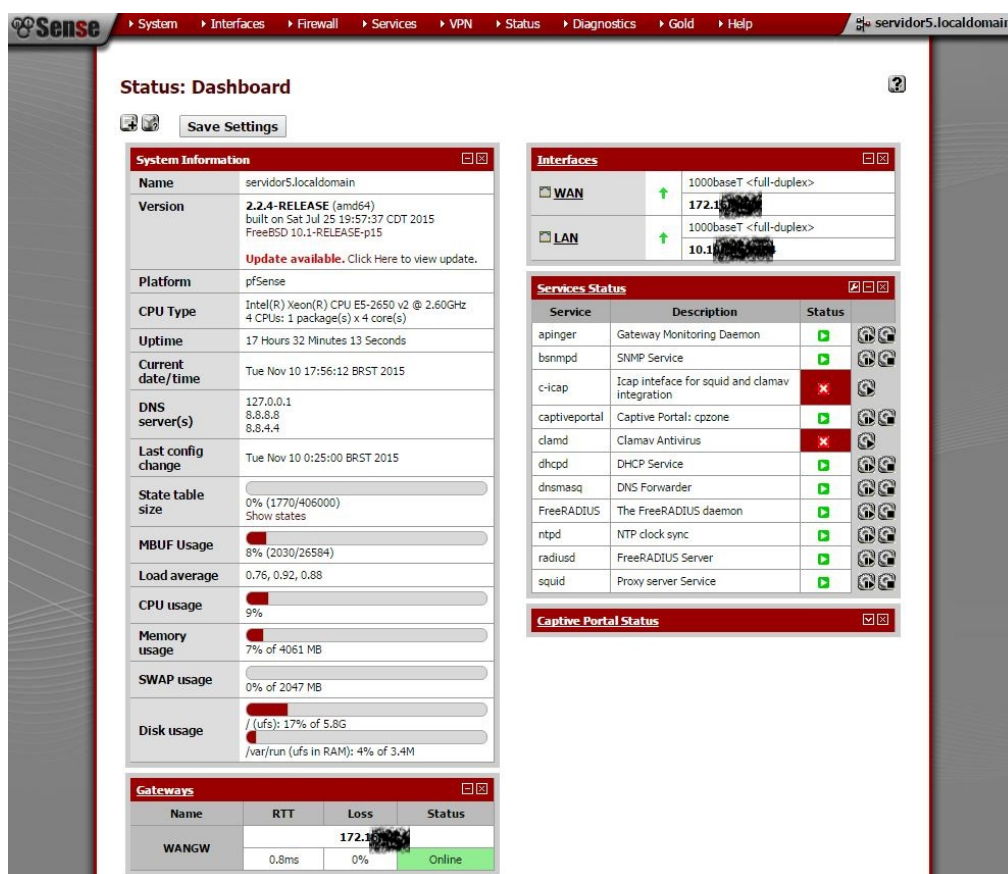


Figura 7.10: Dashboard do PfSense em funcionamento no câmpus Capivari

Fonte: Coordenadoria de Tecnologia da Informação do câmpus

O câmpus possui apenas 1 *access point* para o uso dos usuários finais, mas já existe a projeção de ampliação dessa rede utilizando mais 5 *access point*, melhorando assim a área de cobertura da rede Wi-Fi.

Para acessar a internet o usuário precisa conectar a rede Wi-Fi, em seguida abrir o navegador de internet e utilizar as credenciais para se autenticar no sistema ou o *Voucher* em caso de visitantes. Por fim o próprio sistema redireciona o usuário para a página inicial do site do câmpus Capivari, a partir desse momento o usuário já pode utilizar da internet. A tela de autenticação ao HotSpot pode ser visualizada na Figura 7.10.



**INSTITUTO FEDERAL  
SÃO PAULO  
Campus Capivari**

**Wi-Fi  
INTERNET  
HOTSPOT**

**Seja bem-vindo(a) à area de Wi-Fi do IFSP  
Campus Capivari!**

- Professores, Técnicos Administrativos e Alunos: Use seu usuário e senha.
- Visitante: Use seu voucher.

Se você ainda não tem usuário/senha ou voucher, entre em contato com a Coordenadoria de Tecnologia da Informação e faça seu cadastro.

Para acessar a internet, preencha os dados abaixo:

Usuário:

Senha:

Voucher:

©2013 - Copyright - Desenvolvido por T.I. IFSP Campus Capivari  
 Versão 1.0 - 20/08/2013 - Todos os direitos reservados

*Figura 7.11: Tela de autenticação para uso da rede Wi-Fi do câmpus Capivari*

*Fonte: Coordenadoria de Tecnologia da Informação do câmpus*

Em diversas ocasiões relatadas pela equipe de TI do câmpus, os usuários do serviço Wi-Fi (HotSpot) deslocavam-se até o departamento de informática informando que não estavam conseguindo acesso à internet devido indisponibilidade do sistema.

Para solucionar este tipo de problema eram realizados alguns procedimentos como por exemplo, *restart* do *access point* devido travamento do equipamento, reinício do *FreeRADIUS service* que por motivo ainda desconhecido que deixava de funcionar, houve caso em que foi necessário reiniciar o servidor para normalizar o sistema.

### 7.5.2 Processo de Implementação do Zabbix

O processo de implantação do monitoramento dessa rede se deu através da utilização de um *template* disponibilizado<sup>13</sup> pela comunidade Zabbix, cujo o nome foi mantido. A comunicação do Zabbix Server com o agente monitorado se deu através do protocolo SNMP v2.

Neste *template* já existem aplicações para o monitoramento de diversos itens, dentre eles, processos e *status* do sistema, espaço de disco, memória, interface de rede. Conforme pode ser visto na Figura 7.11. Também foi customizado o monitoramento web no *template*, para monitorar o captive portal do câmpus. Com o uso deste *template* houve ganho de tempo na implementação, bastando apenas customizar para atender as necessidades.



Figura 7.12: Template Pfsense com suas aplicações

Para monitorar a disponibilidade do HotSpot do câmpus Capivari houve a necessidade de criar um cenário web onde é configurado a página a ser acessada, os passos a serem seguidos e os códigos de retorno da página web, conforme a RFC 2616<sup>14</sup>.

A primeira aba (cenário) são definidos o nome, aplicação, intervalo de atualização, quantidade de tentativa e qual navegador para execução dos testes. Na segunda aba (passos), foi criado 2 passos para os testes, no primeiro definimos um nome e a URL do teste, em nosso

13 <https://github.com/ccesario/public/tree/master/zabbix/templates/pfsense/2.1>

14 <http://tools.ietf.org/html/rfc2616#section-10>



caso foi utilizado o endereço que direciona a página de autenticação e o código requerido 200, no segundo passo também definimos outro nome, a URL onde o sistema deve ir após a autenticação e o código requerido que é 200.

Após a configuração do *template* foi criado um *host* virtual, que tem a função de executar os testes de conexão a sites externos, em nosso cenário este *host* é o próprio Zabbix Server associado a este *host* virtual o *template* criado anteriormente.

Por fim foi criado uma *trigger* que monitora a disponibilidade do site, caso o valor de retorno do site seja diferente de “OK” o alerta é disparado com a severidade de nível alto.

### **7.5.3 Benefícios pós implementação**

Com o monitoramento do sistema de HotSpot do câmpus Capivari os técnicos de TI conseguem identificar com rapidez quaisquer incidentes, a quantidade real de incidentes e os momentos das ocorrências destes incidentes.

Para a equipe de TI a quantidade de reclamações quanto ao acessos dessa rede extinguiram, com isso a equipe pode despende seu esforços ao atendimento de outras demandas.

## 8 CONSIDERAÇÕES FINAIS

Durante o desenvolvimento desse trabalho foi possível perceber a grande importância do monitoramento de equipamentos, serviços e sistemas de redes de computadores, sendo ela de pequeno, médio e grande porte. Neste período houve a oportunidade de acompanhar algumas ocorrências relacionados a indisponibilidade de equipamentos, serviços e sistemas.

Ter as informações corretas, possibilita aos administradores de rede tomar decisões coerente e em tempo hábil, minimizando assim o *downtime* de equipamentos e principalmente a ociosidade do usuário final dependente daquele sistema ou serviço.

Com a implantação do Zabbix nas redes do câmpus Capivari, podemos elencar algumas importantes características e benefícios encontrados no decorrer desse trabalho como descrito a seguir:

A automatização dos processos como os realizados nas impressoras, que antes era feito manualmente, fez com que a equipe de TI focassem seus esforços em reais problemas que necessitam de acompanhamento pessoal de um técnico.

A facilidade, agilidade e rapidez em diagnosticar possíveis problemas quanto ao armazenamento dos *backups* de servidores virtuais, visto que essas unidades de armazenamento externo que o câmpus adquiriu em 2013, possuem uma grande limitação de espaço. Isso possibilitou que a equipe gestora estude nova estratégia com os equipamentos atuais, ate a aquisição de novos HD's externos ou ate mesmo uma solução de *Storage*.

As características no processo de implantação do sistema de monitoramento e gerenciamento de redes de computadores Zabbix, podem ser citadas como aspectos positivos e negativos.

Relacionado a natureza positiva da implantação do Zabbix, o sistema é totalmente livre regido sobre licença GPL v2, ou seja, para a instituição de ensino IFSP câmpus Capivari que uma autarquia federal e dependente de verbas públicas é muito importante evitar a oneração financeira com sistemas proprietárias.

Ainda sobre o aspecto positivo, podemos elencar que, o câmpus Capivari não existia nenhuma ferramenta de monitoramento de rede implantada e com esse trabalho possibilitou visualizar o quão importante é o trabalho dos técnicos de TI, para manter o bom funcionamento e sincronismo dos equipamentos que compõe a rede de computadores.

Outro ponto positivo que deve ser citado é relacionado a comunidade brasileira que apoia o Zabbix, tanto em redes sociais quanto em blogs, os usuários deste sistema são muito ativos, sanando dúvidas e sugerindo melhorias e customizações para quem está implementando a ferramenta.

Contudo, podemos dizer que o sistema de monitoramento de rede Zabbix, assegura a equipe gestora de TI do câmpus maior tranquilidade em nível de equipamentos, serviços e sistemas. Pois a automatização de alertas aos incidentes garante a assertividade, pontualidade e principalmente a manutenção preventiva a nível operacional da equipe de TI.

## 8.1 Dificuldades encontradas

Houve dificuldades em alguns períodos de desenvolvimento deste trabalho, a exemplo da curva de aprendizado com o Zabbix, visto que foi uma ferramenta um tanto quanto nova para explorá-la e num primeiro momento para fazer a instalação.

Outra dificuldade é relacionado a conteúdo bibliográfico direcionado ao Zabbix. Foram encontrados poucos livros publicados e apenas 2 livros em português, porém em contrapartida, como dito anteriormente, a comunidade de usuário é muito prestativa e solicita a ajudar.

A maior dificuldade encontrada esta diretamente ligada ao câmpus Capivari. O mesmo tem um problema de instabilidade na rede elétrica de forma que há diversas quedas e surtos. Até mesmo os equipamentos de *nobreak* foram danificados várias vezes, com isso inviabilizou seu uso, ou seja, todos os equipamentos de TI estão ligados diretamente na rede elétrica. Este problema já foi diagnosticado e atestado por engenheiro eletricista, mas infelizmente até a conclusão deste trabalho não há previsão de manutenção na rede elétrica.

Com isso houve um comprometimento muito grande ao desenvolvimento do trabalho, pois com as quedas elétricas o banco de dados corrompia e todas as coletas realizadas não era possível recuperá-las.

Num primeiro momento foi utilizado o banco de dados MySQL versão 5.5 e o Zabbix versão 2.4.3 e sempre que havia os problemas elétricos relatados anteriormente todo o banco era comprometido, houve diversas tentativas para recuperação dos dados coletados porém sem sucesso.

No final do trabalho optou-se em mudar o banco de dados, passando a ser utilizado o

PostgreSQL versão 9.3 e o Zabbix versão 2.4.6, notamos que em três eventos de paradas elétricas houve problemas com o banco de dados porém, o próprio Zabbix sugere um comando (REINDEX) a ser executado no *shell* do PostgreSQL, com isso as tabelas eram corrigidas e todo o sistema retornara a seu perfeito funcionamento.

## 8.2 Trabalhos futuros

Como possíveis trabalhos futuros podemos elencar alguns itens relevantes:

- Implementação do Zabbix Proxy;
- Monitorar outros sistemas operacionais tais como, Windows, HP-UX, toda a família BSD;
- Monitoramento de equipamentos e sistemas utilizando os protocolos IPMI e JMX;
- Implementação de outras API's e pluggins ao Zabbix, como ZabTree no auxílio ao tratamento de eventos, Zabbix-Geo criando mapas dinâmicos utilizando do recurso do *Google Maps* representando assim a localização geográfica dos *hosts* e seus incidentes em tempo real;
- Unificar outras ferramentas de forma que trabalhem em conjunto do Zabbix como por exemplo. Grafana para gerar gráficos mais elaborados, GLPI para gerar chamados automatizados, OCS inventory para o controle de inventários. PUPPET para configurar e automatizar instalações de pacotes e softwares em vários servidores, mantendo a configuração centralizada.
- Implementação do envio de alertas por e-mail, sms, jabber (mensageiro instantâneo).

## 9 REFERÊNCIAS

- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 5462: Confiabilidade e manutenibilidade**. Rio de Janeiro: ABNT, 1994.
- BJORKLUND, M.; SCHOENWAEELDER, J. **RFC 7407: A YANG Data Model for SNMP Configuration**. IETF, dez. 2014.
- CASE, J.; FEDOR, M.; SCHOFFSTALL, M.; DAVIN, C. **RFC 1089: SNMP Over Ethernet**. IETF, fev. 1989.
- CASE, J. D.; FEDOR, M.; SCHOFFSTALL, M. L.; DAVIN, J. **RFC 1157: Simple Network Management Protocol (SNMP)**. IETF, mai. 1990.
- CASE, J.; MUNDY, R.; PARTAIN, D.; STEWART, B. **RFC 3410: Introduction and Applicability Statements for Internet-Standard Management Framework**. IETF, dez. 2002.
- CORREIA, M. F. **Gerência de Redes**. União Educacional de Minas Gerais, 2004. Disponível em: <<http://www.si.lopesgazzabi.com.br/TFC/monografias/MONOGRAFIA%252013011062.pdf>> Acessado em: 02 set. 2015.
- COULOURIS, G; DOLLIMORE, J; KINDBERG, T. **Sistemas Distribuídos: Conceitos e Projetos**. 4.ed. São Paulo: Bookman Companhia Editora, 2007. 780 p.
- DEITEL, H. M; DEITEL, P. J; CHOFFNES, D. R. **Sistemas Operacionais**. 3.ed. São Paulo: Pearson Prentice Hall, 2005. 760 p.
- HEIN, M.; GRIFFITHS, D. **SNMP Versions 1 & 2 Simple Network Management Protocol Theory and Practice**. London: International Thomson Computer Press, 1995.
- HORST, A. H. S; PIRES, A. S; DÉO, A. L. B. **De A a ZABBIX**. 1.ed. São Paulo: Novatec Editora Ltda, fev. 2015. 411 p.
- INSTITUTO FEDERAL DE SÃO PAULO. **O Instituto Federal de São Paulo**. Disponível em: <<http://www.ifsp.edu.br/index.php/instituicao/ifsp.html>> Acessado em: 25 out. 2015.
- INSTITUTO FEDERAL DE SÃO PAULO CAMPUS CAPIVARI. **O campus**. Disponível em: <<http://www.ifspcapivari.com.br/ifsp-capivari/>> Acessado em: 25 out. 2015.
- KUROSE, F; ROSS, K. W. **Rede de Computadores e a Internet: uma abordagem top-down**. 5.ed. São Paulo: Pearson Education, 2010. 614 p.
- LIMA, J. R.; **Monitorando interface IPMI**. jul. 2012. Disponível em: <<http://www.vivaolinux.com.br/artigo/Monitorando-interface-IPMI>> Acessado em: 12 out. 2015.

- LIMA, M. M. A. E. **Introdução a Gerenciamento de Redes TCP/IP**. Disponível em: <<https://memoria.rnp.br/newsgen/9708/n3-2.html>> Acessado em: 4 ago. 2015.
- MACHADO, F. B; MAIA, L. P. **Arquitetura de Sistemas Operacionais**. 4.ed. Rio de Janeiro: Livros Técnicos e Científicos Editora S.A., 2007. 304 p.
- MOURA. D. R.; SCHMIDT, K. **Essential SNMP**. 2.ed. Editora O'Reilly, 2005. 462 p.
- ONE LINEA TELECOM, “**Boletim Técnico One Linea.**” Disponível em: <<http://www.onelinea.com.br/pdfs/bto-gerenciamento-redes.pdf>> Acessado em: 13 ago. 2015
- SAYDAM, T.; MAGENDAZ, T. "**From networks and network Management into Service Management.**" Journal of Networks and Systems Management, vol.4, n. 4, dez 1996, p. 345-348.
- SMITH, R. et. al. **RFC 1759: Print MIB**. IETF, mar. 1995.
- SNMP RESEARCH INTERNATIONAL, INC. **SNMP RFCs**. Disponível em: <[http://www.snmp.com/protocol/snmp\\_rfcs.shtml](http://www.snmp.com/protocol/snmp_rfcs.shtml)> Acessado em: 14 set. 2015
- STALLINGS, W. **Redes e Sistemas de Comunicação de Dados: Teoria e Aplicações Corporativas**. 5. ed. Rio de Janeiro: Elsevier, 2005.
- TANENBAUM, Andrew S. **Redes de Computadores**. 4.ed. Editora Campus (Elsevier), 2003.
- TORRES, G. **Redes de Computadores Curso Completo**. 1.ed. Rio de Janeiro: Axcel Books do Brasil Editora, 2001. 664p.

## APÊNDICE A – INSTALAÇÃO DO ZABBIX SERVER 2.4 COM POSTGRESQL 9.3

```
# Variáveis para facilitar a instalação
export VERSAO="2.4.7";
export SOURCE_DIR="/install/zabbix-$VERSAO";
export SENHA="zabbix";
export SENHAROOT="zabbix_root";
export NOMBANCO="zbx_db";
export USUARIODB="zbx_user"
export ZABBIXCONF="/usr/local/etc/zabbix_server.conf"
export PHPINI="/etc/php.ini";
export POSTGRESQL="/var/lib/pgsql/9.3/data/postgresql.conf";
export PG_HBA="/var/lib/pgsql/9.3/data/pg_hba.conf";

# Repositório EPEL
wget http://epel.gtinternet.com/6/x86_64/epel-release-6-8.noarch.rpm
rpm -ivh epel-release-6*.rpm

# Instalar ferramentas de desenvolvimento
yum -y install groupinstall 'Development Tools'
yum -y install zlib-devel glibc-devel curl-devel automake libidn-devel openssl-devel
yum -y install rpm-devel

# Pré-requisito - SNMP
yum -y install net-snmp net-snmp-devel net-snmp-utils

# Pré-requisito - Sistema Operacional
yum -y install OpenIPMI OpenIPMI-devel libssh2-devel make fping yum-utils

# Pré-requisito - INTERFACE WEB
yum -y install php php-bcmath php-gd php-mbstring php-xml php-ldap httpd php-curl php-pgsql

# Pré-requisito - BANCO DE DADOS PostgreSql
rpm -Uvh http://yum.postgresql.org/9.3/redhat/rhel-6-x86_64/pgdg-centos93-9.3-1.noarch.rpm
yum install -y postgresql93-server postgresql93-contrib postgresql93 postgresql93-devel
php-pgsql nodejs-pg

# Atualizar tudo
yum -y update

# Criar usuário e grupo zabbix para o sistema operacional
groupadd zabbix
useradd -g zabbix zabbix

# Inicializar o PostgreSQL
```

```
service postgresql-9.3 initdb
```

```
# Colocar PostgreSQL na inicialização automática
```

```
chkconfig --add postgresql-9.3
```

```
chkconfig postgresql-9.3 on
```

```
#Backup das configurações originais do PostgreSQL
```

```
cp -Rap $POSTGRESQL $POSTGRESQL.bkp
```

```
cp -Rap $PG_HBA $PG_HBA.bkp
```

```
# Configurar o POSTGRESQL.CONF
```

```
echo "listen_addresses = 'localhost'" >> $POSTGRESQL;
```

```
echo "port = 5432" >> $POSTGRESQL;
```

```
echo "max_connections = 100" >> $POSTGRESQL;
```

```
echo "shared_buffers = 128MB" >> $POSTGRESQL;
```

```
echo "dynamic_shared_memory_type = posix" >> $POSTGRESQL;
```

```
echo "log_destination = 'stderr'" >> $POSTGRESQL;
```

```
echo "logging_collector = on" >> $POSTGRESQL;
```

```
echo "log_directory = 'pg_log'" >> $POSTGRESQL;
```

```
echo "log_filename = 'postgresql-%a.log'" >> $POSTGRESQL;
```

```
echo "log_truncate_on_rotation = on" >> $POSTGRESQL;
```

```
echo "log_rotation_age = 1d" >> $POSTGRESQL;
```

```
echo "log_rotation_size = 0" >> $POSTGRESQL;
```

```
echo "log_line_prefix = '< %m >'" >> $POSTGRESQL;
```

```
echo "log_timezone = 'Brazil/East'" >> $POSTGRESQL;
```

```
echo "datestyle = 'iso, dmy'" >> $POSTGRESQL;
```

```
echo "timezone = 'Brazil/East'" >> $POSTGRESQL;
```

```
echo "lc_messages = 'pt_BR.UTF-8'" >> $POSTGRESQL;
```

```
echo "lc_monetary = 'pt_BR.UTF-8'" >> $POSTGRESQL;
```

```
echo "lc_numeric = 'pt_BR.UTF-8'" >> $POSTGRESQL;
```

```
echo "lc_time = 'pt_BR.UTF-8'" >> $POSTGRESQL;
```

```
echo "default_text_search_config = 'pg_catalog.portuguese'" >> $POSTGRESQL;
```

```
# Configurar o PG_HBA
```

```
echo "host all all trust" > $PG_HBA;
```

```
echo "host all all 127.0.0.1/32 trust" >> $PG_HBA;
```

```
echo "host all all ::1/128 trust" >> $PG_HBA;
```

```
echo "host zbx_db zbx_user 127.0.0.1/32 trust" >> $PG_HBA;
```

```
# Reiniciar o serviço de banco de dados
```

```
service postgresql-9.3 restart
```

```
# Acessar o banco de dados via PSQL
```

```
su postgres
```

```
psql
```

```
#Criar o usuário zbx_user no banco de dados
```

```
CREATE USER zbx_user;
```

```
# Adicionando a senha para usuário do banco de dados
```



```

ALTER USER zbx_user WITH PASSWORD 'zabbix';

# Criar o banco de dados
CREATE DATABASE zbx_db;

# Garantir privilégio de acesso ao banco de dados
GRANT ALL PRIVILEGES ON DATABASE zbx_db TO zbx_user;

# Alterar o owner do banco zbx_db para o usuario zbx_user
ALTER DATABASE zbx_db OWNER TO zbx_user;

# Sair do prompt do banco de dados
\q
exit

# Inicializar o apache
service httpd start

# Colocar o apache na inicialização automática
chkconfig --add httpd
chkconfig httpd on

# Backup das configurações do PHP.INI
cp -Rap $PHPINI $PHPINI.bkp

# Configurando o PHP.INI
sed -i "s/date.timezone;/date.timezone/" $PHPINI;
sed -i "s/max_execution_time;/max_execution_time/" $PHPINI;
sed -i "s/max_input_time;/max_input_time/" $PHPINI;
sed -i "s/post_max_size;/post_max_size/" $PHPINI;

# Atualizando os pré-requisitos do zabbix no php.ini
echo "date.timezone =America/Sao_Paulo" >> $PHPINI;
echo "max_execution_time = 300" >> $PHPINI;
echo "max_input_time = 300" >> $PHPINI;
echo "post_max_size = 16M" >> $PHPINI;
echo "always_populate_raw_post_data=-1" >> $PHPINI;

# Reiniciar o apache
service httpd restart

# Criar diretório INSTALL e acessá-lo
mkdir /install
cd /install

# Baixar o arquivo do zabbix na pasta INSTALL
wget http://downloads.sourceforge.net/project/zabbix/ZABBIX%20Latest%20Stable/
$VERSAO/zabbix-$VERSAO.tar.gz

# Descompactar o arquivo

```

```
tar xzvf zabbix-$VERSAO.tar.gz
```

```
# Entrar na pasta descompactada
cd zabbix-$VERSAO
```

```
# Copiar dados para o zbx_db
psql -U zbx_user -W zbx_db < $SOURCE_DIR/database/postgresql/schema.sql
psql -U zbx_user -W zbx_db < $SOURCE_DIR/database/postgresql/images.sql
psql -U zbx_user -W zbx_db < $SOURCE_DIR/database/postgresql/data.sql
```

```
#instalação da GUI do Zabbix
cp -Rpv frontends/php/* /var/www/html/
```

```
# Compilar os binários (Servidor Zabbix com POSTGRESQL + Agente)
./configure --enable-server --enable-agent --with-postgresql=/usr/pgsql-9.3/bin/pg_config
--with-net-snmp --with-libcurl --with-libxml2 --with-openipmi && make install
```

```
# Backup dos arquivos de configuração do Zabbix Server
cp /usr/local/etc/zabbix_server.conf /usr/local/etc/zabbix_server.conf.bkp
```

```
# Parâmetros de configuração do Zabbix Server
echo "DBHost=127.0.0.1" > $ZABBIXCONF
echo "DBPort=5432" >> $ZABBIXCONF
echo "DBName=$NOMEBANCO" >> $ZABBIXCONF
echo "DBUser=$USUARIODB" > $ZABBIXCONF
echo "DBPassword=$SENHA" >> $ZABBIXCONF
echo "StartPollers=8" >> $ZABBIXCONF
echo "StartPollersUnreachable=8" >> $ZABBIXCONF
echo "StartTrappers=8" >> $ZABBIXCONF
echo "StartPingers=4" >> $ZABBIXCONF
echo "StartDiscoverers=2" >> $ZABBIXCONF
echo "StartHTTTPollers=2" >> $ZABBIXCONF
echo "HousekeepingFrequency=24" >> $ZABBIXCONF
echo "MaxHousekeeperDelete=10000" >> $ZABBIXCONF
echo "StartDBSyncers=5" >> $ZABBIXCONF
echo "HistoryCacheSize=32M" >> $ZABBIXCONF
echo "TrendCacheSize=32M" >> $ZABBIXCONF
echo "HistoryTextCacheSize=64M" >> $ZABBIXCONF
echo "CacheSize=32M" >> $ZABBIXCONF
echo "DebugLevel=3" >> $ZABBIXCONF
echo "PidFile=/tmp/zabbix_server.pid" >> $ZABBIXCONF
echo "LogFileSize=10" >> $ZABBIXCONF
echo "LogFile=/tmp/zabbix_server.log" >> $ZABBIXCONF
echo "Timeout=3" >> $ZABBIXCONF
```

```
# Copiar o arquivo de inicialização do servidor Zabbix
cp $SOURCE_DIR/misc/init.d/fedora/core5/zabbix_server /etc/init.d
chkconfig --add zabbix_server
chkconfig zabbix_server on
```

```
#inicializando o zabbix server  
service zabbix_server start
```

```
# Verificar se o server esta rodando  
tail /tmp/zabbix_server.log
```

```
# Copiar o arquivo de inicialização do Zabbix Agent  
cp $SOURCE_DIR/misc/init.d/fedora/core5/zabbix_agentd /etc/init.d  
chkconfig --add zabbix_agentd  
chkconfig zabbix_agentd on
```

```
# Inicializando o zabbix server  
service zabbix_agentd start
```

```
# Desabilitando SELinux (apenas para teste, em produção deve ser devidamente configurado)  
echo "SELINUX=disabled" > /etc/selinux/config  
echo "SELINUXTYPE=targeted" >> /etc/selinux/config
```

```
# Firewall desabilitado (apenas para teste, em produção deve ser devidamente configurado)  
service iptables stop
```

## APÊNDICE B – CONCLUSÃO DA INSTALAÇÃO DO ZABBIX 2.4 VIA *BROWSER*

Usando um navegador web, acessar o endereço IP do Zabbix Server (<http://ipdozabbix>)



Figura 1: Tela de boas vindas a instalação do Zabbix Server

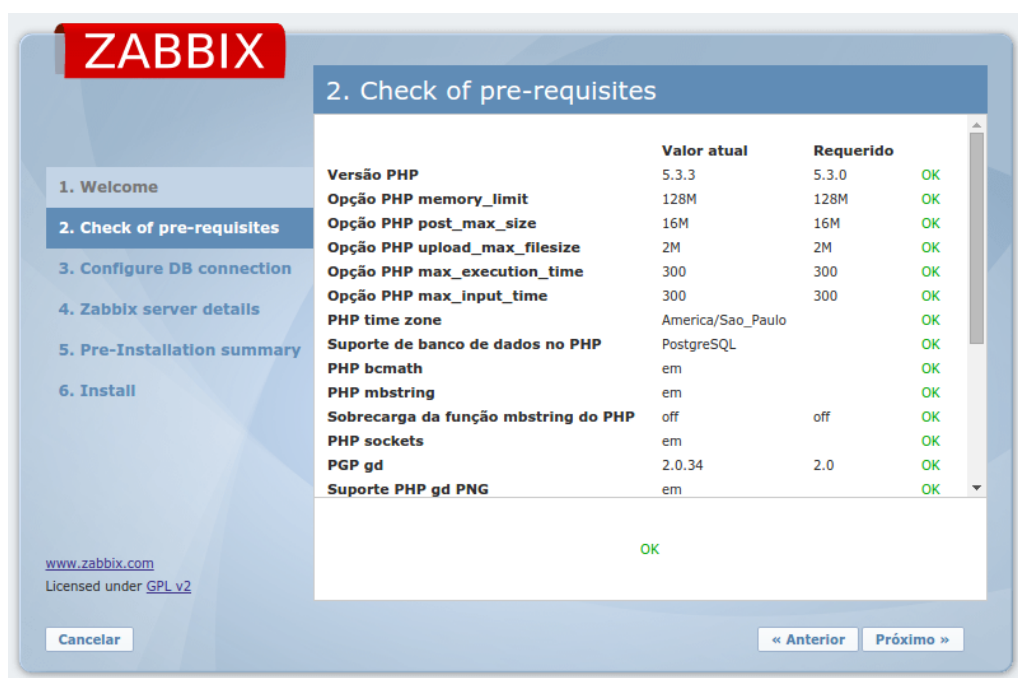


Figura 2: Tela de checagem de parâmetros e dependências de pacotes



**ZABBIX**

### 3. Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database.

Press "Test connection" button when done.


<b>Tipo de banco de dados</b>	PostgreSQL ▼
<b>Host do banco de dados</b>	localhost
<b>Porta do banco de dados</b>	0 0 - use default port
<b>Nome do banco de dados</b>	zbx_db
<b>Esquema do banco de dados</b>	
<b>Usuário</b>	zbx_user
<b>Senha</b>	*****

[www.zabbix.com](#)  
Licensed under [GPL v2](#)

[Cancelar](#) [« Anterior](#) [Próximo »](#)

[Test connection](#)

*Figura 3: Tela de configuração da conexão com banco de dados*



**ZABBIX**

### 3. Configure DB connection

Please create database manually, and set the configuration parameters for connection to this database.

Press "Test connection" button when done.

<b>Tipo de banco de dados</b>	PostgreSQL ▼
<b>Host do banco de dados</b>	localhost
<b>Porta do banco de dados</b>	0 0 - use default port
<b>Nome do banco de dados</b>	zbx_db
<b>Esquema do banco de dados</b>	
<b>Usuário</b>	zbx_user
<b>Senha</b>	*****

[www.zabbix.com](#)  
Licensed under [GPL v2](#)

[Cancelar](#) [« Anterior](#) [Próximo »](#)

[Test connection](#)

OK

*Figura 4: Tela de teste de conexão para habilitar próxima tela*



## 4. Zabbix server details

Please enter host name or host IP address and port number of Zabbix server, as well as the name of the installation (optional).

Host:

Porta:

Nome:

[www.zabbix.com](http://www.zabbix.com)  
Licensed under [GPL v2](#)

Figura 5: Tela para definição da porta e nome do servidor



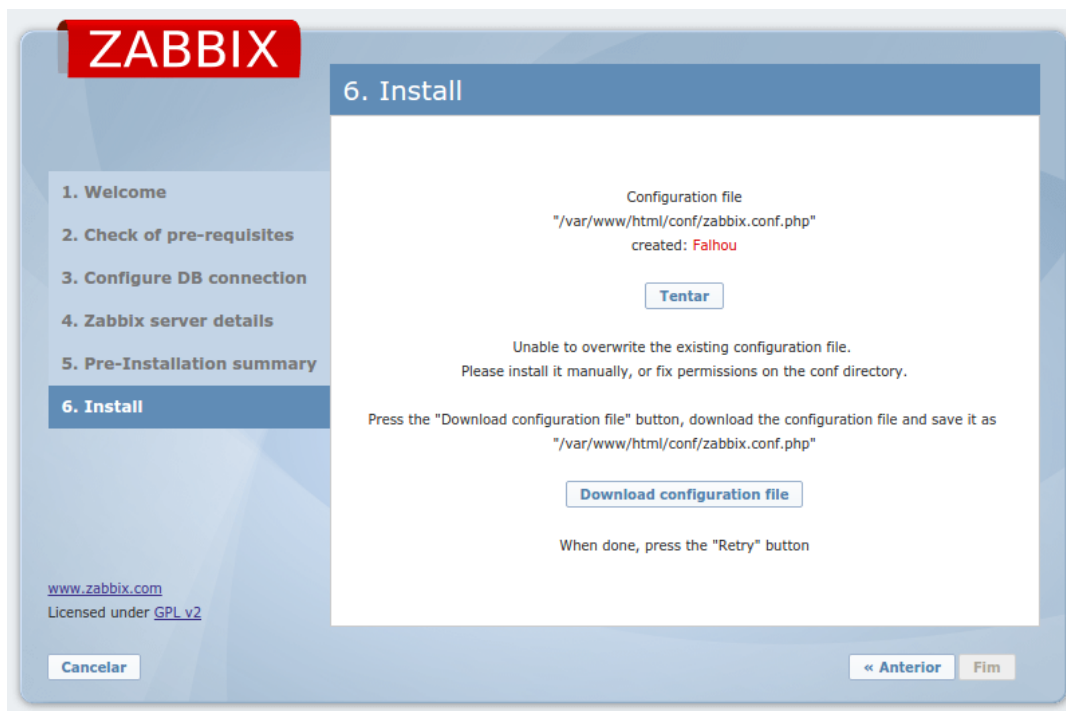
## 5. Pre-Installation summary

Please check configuration parameters.  
If all is correct, press "Next" button, or "Previous" button to change configuration parameters.

<b>Tipo de banco de dados</b>	PostgreSQL
<b>Servidor de banco de dados</b>	localhost
<b>Porta do banco de dados</b>	padrão
<b>Nome do banco de dados</b>	zbx_db
<b>Usuário do banco de dados</b>	zbx_user
<b>Senha do banco de dados</b>	*****
<b>Esquema do banco de dados</b>	
<b>Servidor Zabbix</b>	localhost
<b>Porta do servidor Zabbix</b>	10051
<b>Nome do servidor Zabbix</b>	

[www.zabbix.com](http://www.zabbix.com)  
Licensed under [GPL v2](#)

Figura 6: Tela para checagem final dos parâmetros de configuração



*Figura 7: Tela de finalização de instalação*

## APÊNDICE C – INSTALAÇÃO DO ZABBIX AGENT 2.4.7 EM LINUX

```
# Variáveis para a instalação
export VERSAO="2.4.7";
export SOURCE_DIR="/install/zabbix-$VERSAO";
export ZAGENT="/usr/local/etc/zabbix_agentd.conf";

# Cria usuário Zabbix sem shell
groupadd zabbix
useradd -g zabbix zabbix

# Cria diretório INSTALL e acessa
mkdir /install
cd /install

# Baixa o arquivo na pasta INSTALL
wget http://downloads.sourceforge.net/project/zabbix/ZABBIX%20Latest%20Stable/
$VERSAO/zabbix-$VERSAO.tar.gz

# Descompacta o arquivo
tar xzvf zabbix-$VERSAO.tar.gz

# Entra na pasta descompactada
cd zabbix-$VERSAO

# Criar diretório ZABBIX e copiar os arquivos necessários
mkdir /etc/zabbix
cp $SOURCE_DIR /bin/* /etc/zabbix
cp $SOURCE_DIR /sbin/* /etc/zabbix
chown -R zabbix:zabbix /etc/zabbix

# copia ZABBIX_AGENTD
cp $SOURCE_DIR/misc/init.d/fedora/core5/zabbix_agentd /etc/init.d

#Inicialização do servidor Zabbix automática
chkconfig --add zabbix_agentd
chkconfig --level 35 zabbix_agentd on

#Backup dos arquivos de configuração
cp /usr/local/etc/zabbix_agentd.conf /usr/local/etc/zabbix_agentd.conf.bkp

#configurando o ZABBIX AGENTD
echo "LogFile=/tmp/zabbix_agentd.log" > $ZAGENT;
echo "Server=IP_ZABBIX_SERVER" >> $ZAGENT;
echo "ListenPort=10050" >> $ZAGENT;
echo "StartAgents=3" >> $ZAGENT;
```



```
echo "ServerActive=IP_ZABBIX_SERVER" >> $ZAGENT;  
echo "Timeout=3" >> $ZAGENT;  
echo "Hostname=NOME_DO_HOST_AGENT" >> $ZAGENT;
```

```
#inicializando o zabbix server  
service zabbix_agentd start
```