

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

CARLOS HENRIQUE GARCIA DA SILVA

MONITORAMENTO DE AMBIENTE DE REDE COM A FERRAMENTA
ZABBIX

TRABALHO DE CONCLUSÃO DE CURSO

PONTA GROSSA

2021

CARLOS HENRIQUE GARCIA DA SILVA

**MONITORAMENTO DE AMBIENTE DE REDE COM A FERRAMENTA
ZABBIX**

NETWORK ENVIRONMENT MONITORING WITH THE ZABBIX TOOL

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas, do Departamento Acadêmico de Informática, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof^a. Dra. Helyane Bronoski Borges

PONTA GROSSA

2021



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Ponta Grossa



CARLOS HENRIQUE GARCIA DA SILVA

MONITORAMENTO DE AMBIENTE DE REDE COM A FERRAMENTA ZABBIX

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas da Universidade Tecnológica Federal do Paraná (UTFPR).

Data de aprovação: 18 de maio de 2021.

Profa. Helyane Bronoski Borges, Doutorado - Universidade Tecnológica Federal do Paraná
Prof. Augusto Foronda, Doutorado - Universidade Federal do Paraná
Prof. Geraldo Ranthum, Mestrado - Universidade Tecnológica Federal do Paraná

AGRADECIMENTOS

Primeiramente agradeço a Deus, pela minha vida, e por me ajudar a ultrapassar todos obstáculos encontrados ao longo do curso.

Agradeço a meus familiares, principalmente meus pais que sempre estiveram presentes incentivando e apoiando para atingir meus objetivos.

Agradeço a meus amigos, por todos momentos de apoio e incentivo.

Agradeço a minha orientadora Helyane Borges, pela paciência para me guiar nesta trajetória.

Por fim aos professores da UTFPR, pelos ensinamentos que contribuíram minha formação profissional.

No meio da dificuldade encontra-se a
oportunidade.
(Albert Einstein)

RESUMO

Este trabalho pretende demonstrar a solução para monitoramento de rede utilizando a ferramenta Zabbix, integrado ao Grafana e Telegram. A estrutura de rede consiste em um servidor de gerenciamento principal, onde será instalado o Servidor Zabbix, e os hosts que serão monitorados, sendo um Servidor Web, um Servidor AD/DNS, um Servidor Zabbix e um Roteador. O Servidor Zabbix fará o monitoramento da estrutura, integrado ao Telegram, que irá disparar informações para o administrador da rede sobre o funcionamento do ambiente monitorado. O Grafana possibilitará a demonstração de recursos do ambiente a fim de proporcionar estratégia de negócio e ações preditivas. Será documentada a instalação, configuração e monitoramento da estrutura. O objetivo é mostrar que é possível monitorar o ambiente de rede utilizando a ferramenta Zabbix de forma efetiva.

Palavras-chave: Zabbix. Monitoramento. Gerenciamento. Grafana. Telegram.

ABSTRACT

This work intends to demonstrate the solution for network monitoring using the Zabbix tool, linked to Grafana and Telegram. The network structure consists of a main management server, where the Zabbix Server will be installed, and the hosts that will be monitored, being a Web Server, an AD / DNS Server, a Zabbix Server and a Router. The Zabbix Server will monitor the structure, integrated with Telegram will trigger information for the network administrator about the operation of the monitored environment. Grafana had enabled the demonstration of environmental resources in order to provide business strategy and predictive actions. The installation, configuration and monitoring of the structure will be documented. The goal is to show that it is possible to monitor the network environment effectively.

Keywords: Zabbix. Monitoring. Management. Grafana. Telegram.

LISTA DE ILUSTRAÇÕES

Figura 1 - Agente e Gerente SNMP	17
Figura 2 - Dashboard Grafana.....	20
Figura 3 - Dashboard Zabbix.....	21
Figura 4 - Arquitetura Implementada	25
Figura 5 - Ativando Plugin do Zabbix no Grafana.....	26
Figura 6 - Adicionando Conexão do Zabbix no Grafana	27
Figura 7 - Adicionando Conexão Zabbix no Grafana 2	28
Figura 8 - Adicionando Conexão Mysql no Grafana	29
Figura 9 - Adicionando Conexão Mysql no Grafana 2.....	29
Figura 10 - Adicionando Conexão Mysql no Grafana 3.....	30
Figura 11 - Criando Robô no Telegram.....	30
Figura 12 - Criando Robô no Telegram 2.....	31
Figura 13 - Integrando Robô ao Zabbix.....	32
Figura 14 - Criando Novo Grupo no Telegram	32
Figura 15 - Adicionando Robô ao Grupo no Telegram.....	33
Figura 16 - Localizando ID do Grupo do Telegram	33
Figura 17 – Adicionando Grupo do Telegram ao Zabbix.....	34
Figura 18 - Criando Nova Ação de Envio Para o Telegram	35
Figura 19 - Ação de Envio Para o Telegram - Operações.....	36
Figura 20 - Ação de Envio Para o Telegram - Recuperação.....	37
Figura 21 - Ação de Envio Para o Telegram - Atualização.....	38
Figura 22 - Tela de Configuração de Hosts.....	39
Figura 23 - Criação de Hosts.....	39
Figura 24 - Adicionando <i>Templates</i> aos <i>Hosts</i>	40
Figura 25 - Dashboard Zabbix.....	42
Figura 26 - Interface dos Hosts Monitorados.....	42
Figura 27 - Teste de Parada Serviço Apache.....	43
Figura 28 - Incidentes Recebidos Telegram.....	43
Figura 29 – Teste de Inicio Serviço Apache.....	44
Figura 30 – <i>Dashboard</i> Configurado Grafana	45
Figura 31 - Obtendo Repositório Zabbix	51
Figura 32 - Instalando repositório do Zabbix	52
Figura 33 - Atualizando Lista de Repositórios Zabbix	52
Figura 34 - Instalação dos Pacotes Zabbix	52
Figura 35 - Instalação do servidor de Banco de dados MariaDB	53
Figura 36 - Ajuste Arquivo de Configuração Zabbix	54
Figura 37 - Ajuste Arquivo de Configuração Apache.....	55
Figura 38 - Iniciando Zabbix Server, Agente Zabbix e Apache	55

Figura 39 - Interface Web Instalador Zabbix	56
Figura 40 - Tela Checagem de Pré-requisitos	57
Figura 41 - Configuração da Conexão com o Banco de Dados	58
Figura 42 - Detalhes da Configuração do Servidor	58
Figura 43 - Confirmação da Instalação	59
Figura 44 - Tela de Login do Zabbix.....	59
Figura 45 - Instalação dos Pacotes Grafana	61
Figura 46 - Adicionando Repositório Grafana	61
Figura 47 - Atualizando Repositórios e Instalando o Grafana	62
Figura 48 – Instalação do <i>Plugin</i> de Integração do Grafana com Zabbix.....	62
Figura 49 - Adicionando Grafana Início Automático, Ativando e Inicializando.	63
Figura 50 - Tela de Login do Grafana	63
Quadro 1 - Versões Protocolo SNMP.....	17
Quadro 2 - Itens Monitorados Pelo Zabbix.....	41

SUMÁRIO

1 INTRODUÇÃO	11
1.1 OBJETIVOS	12
1.1.1 Objetivo Geral	12
1.1.2 Objetivos Específicos	12
1.2 JUSTIFICATIVA	12
1.3 ORGANIZAÇÃO DO TRABALHO	13
2 REFERENCIAL TEÓRICO	14
2.1 GERENCIAMENTO DE REDE	14
2.2 PROTOCOLO SNMP	15
2.2.1 MIB	15
2.2.2 Agentes e Gerentes	16
2.2.3 Operações SNMP	16
2.2.4 Versões SNMP	17
2.3 SOFTWARE ZABBIX	18
2.3.1 Arquitetura Zabbix	18
2.4 GRAFANA	19
2.5 TELEGRAM	21
2.6 TRABALHOS RELACIONADOS	22
3 IMPLEMENTAÇÃO DO MONITORAMENTO DE REDE	24
3.1 ARQUITETURA DE REDE	24
3.2 ETAPAS DO DESENVOLVIMENTO DO TRABALHO	25
3.2.1 Etapa 1 - Instalação e Configuração da Ferramenta Zabbix	25
3.2.2 Etapa 2 - Instalação e Integração do Grafana	26
3.2.3 Etapa 3 – Integração do Telegram	30
3.2.4 Etapa 4 – Configuração do Monitoramento de Rede	38
3.2.5 Etapa 5 – Testes do Monitoramento de Rede	40
3.2.6 Etapa 6 – Análise dos Testes de Monitoramento de Rede	40
4 TESTES E RESULTADOS	41
4.1 DESCRIÇÃO DO AMBIENTE TESTADO	41
4.2 TESTES ZABBIX E TELEGRAM	42
4.3 TESTES ZABBIX E GRAFANA	44
4.4 RESULTADOS OBTIDOS COM OS TESTES	45
5 CONCLUSÃO	47
5.1 TRABALHOS FUTUROS	47
REFERÊNCIAS	48
APÊNDICE A – Instalação e Configuração da ferramenta Zabbix e da aplicação Grafana	50

1 INTRODUÇÃO

O uso de dispositivos computacionais tem-se tornado parte do cotidiano da sociedade. De acordo com a previsão realizada em 2020 pela Cisco, acredita-se que em 2023 serão 29,3 bilhões de dispositivos conectados à internet, número que representa um total de 3,6 dispositivos por pessoa no mundo (CISCO, 2020).

Com o aumento da variedade de novas tecnologias, o número de equipamentos e serviços utilizados para otimizar funções e auxiliar na produtividade de colaboradores nas empresas, tem aumentado consideravelmente. O setor de Tecnologia da Informação (TI), se tornou o principal responsável pelas atualizações tecnológicas nas empresas. Para garantir a qualidade do serviço prestado pelos profissionais da área de TI, o monitoramento de rede, que ocupa grande quantidade de horas do setor, necessita de ferramentas que auxiliem no gerenciamento da rede (FONSECA, 2018).

Entre as razões que tornam o gerenciamento da rede uma tarefa difícil está a heterogeneidade da Intranet na maioria das organizações. O segundo ponto de atenção é o avanço da tecnologia, a necessidade de constante atualização de produtos e serviços e por último é o tamanho das redes, onde a distância pode dificultar a comunicação remota entre dispositivos. O gerenciamento de rede não deve se limitar ao monitoramento do funcionamento dos dispositivos, e sim proporcionar ao administrador de rede ferramentas que possibilitem antecipar problemas que possam causar a indisponibilidade de serviços e da rede (BAHLS, 2016).

O Zabbix é uma ferramenta de código aberto, utilizada por grande parte das corporações e possibilita que até mesmo as empresas de pequeno porte possam monitorar o seu ambiente de rede. A ferramenta, proporciona um monitoramento completo do ambiente de rede, visto que qualquer dispositivo que possua o protocolo SNMP (*Simple Network Management Protocol*) pode ser monitorado através da coleta de dados realizada no dispositivo. Ainda, é possível fazer monitoramento dos serviços e recursos utilizados nos *hosts* monitorados, possibilitando a programação de alertas e disparo de incidentes, informado de forma instantânea o administrador da rede (ZABBIX, 2021).

O objetivo deste trabalho é implementar um ambiente com monitoramento de rede. Para isto será utilizado a ferramenta Zabbix, em virtude da sua capacidade de coletar e diagnosticar eventuais alertas e incidentes na rede. O Zabbix será integrado

ao Grafana e Telegram para proporcionar um ambiente de rede reativo, mantendo a alta disponibilidade de seus serviços.

1.1 OBJETIVOS

Os objetivos deste trabalho são descritos a seguir.

1.1.1 Objetivo Geral

Monitorar o ambiente de rede com o software Zabbix, integrado com Grafana e Telegram.

1.1.2 Objetivos Específicos

- Configurar o Servidor Zabbix;
- Configurar o Agente do Zabbix nos servidores que serão monitorados: Servidor Zabbix, Servidor AD e Servidor Apache;
- Integrar a ferramenta Zabbix com as aplicações Grafana e Telegram para auxílio no monitoramento de rede;
- Realizar testes do sistema de monitoramento Zabbix.
- Analisar as informações obtidas da ferramenta Zabbix.

1.2 JUSTIFICATIVA

O monitoramento ambiente de rede, contribui para um ambiente com baixa indisponibilidade, os alertas são essenciais para o administrador de rede, pois, possibilita a ação antes que os erros ocorram.

Monitorar os *hosts* possibilita a tratativa de erros de forma rápida, pois os incidentes são disparados imediatamente após a falha, impactando diretamente na qualidade de serviço fornecido pelo setor de TI.

O Grafana integrado ao Zabbix possibilita que o administrador de rede possa monitorar visualmente a qualidade do ambiente de rede, permitindo a intervenção antes que incidentes possam ocorrer.

A qualidade do ambiente rede em uma empresa, está diretamente ligada ao desempenho de suas ferramentas e disponibilidade de seus serviços, um ambiente monitorado e configurado de forma eficiente irá tornar a rede confiável e com alta disponibilidade.

1.3 ORGANIZAÇÃO DO TRABALHO

O trabalho está dividido em 5 Capítulos. O Capítulo 2 apresenta a fundamentação teórica utilizada no desenvolvimento do trabalho. O Capítulo 3 apresenta a arquitetura do ambiente de rede a ser monitorado e todas etapas para chegar no completo monitoramento de rede. O Capítulo 4 apresenta os testes realizados no ambiente monitorado e os resultados obtidos. E por fim, o Capítulo 5 descreve a conclusão do trabalho.

2 REFERENCIAL TEÓRICO

Este Capítulo apresenta a fundamentação teórica necessária para o desenvolvimento e entendimento deste trabalho. A Seção 2.1 aborda conceitos de gerenciamento de rede. A Seção 2.2 apresenta uma visão completa do protocolo SNMP. A Seção 2.3 apresenta o software Zabbix e suas características. Por fim, na Seção 2.4 e 2.5 é feita uma breve apresentação do software Grafana e Telegram, respectivamente.

2.1 GERENCIAMENTO DE REDE

O gerenciamento de rede visa assegurar certo nível de qualidade dos serviços. Lopes (2003, p. 17) considera que, um sistema de gerenciamento de redes pode ser definido como um conjunto de ferramentas integradas para monitoramento e controle da rede.

Os sistemas de gerenciamento de rede possuem em comum quatro componentes básicos: elementos gerenciados, estação de gerência, protocolo de gerência e informações de gerência. Os elementos gerenciados, possuem o *software* agente, este sendo responsável por permitir que a estação seja monitorada e fornecer o controle para a estação de gerência. A estação de gerência é o *software*, denominado “Gerente”, que faz o controle ou monitoramento dos elementos gerenciados. O *software* executará a comunicação direta com os agentes nos elementos gerenciados e é por meio da estação de gerência que os usuários autorizados possuem acesso para gerenciar a rede (LOPES; SAUVE; NICOLLETTI, 2003, p. 17).

Para que ocorra a comunicação de forma efetiva entre elementos gerenciados e a estação de gerência, há necessidade de um protocolo, este por sua vez chamado de protocolo de gerência. Esse protocolo permite operações de monitoramento e controle dos elementos gerenciados. O gerente e os agentes trocam informações através do protocolo de gerência, no entanto, não é qualquer tipo informação transmitida entre eles. As informações de gerência definem quais são os dados que os elementos gerenciados podem transmitir para a estação de gerência (LOPES; SAUVE; NICOLLETTI, 2003, p. 17).

2.2 PROTOCOLO SNMP

O Protocolo de Gerenciamento de Rede Simples (SNMP), trata de um conjunto de operações simples para gerenciamento de rede, possibilitando a coleta de dados em dispositivos e a alteração do estado destas informações (BONOMO, 2006).

A padronização de solução de gerência mais usada no mundo chama-se *Internet-Standard Network Management Framework*. Esta solução é mais conhecida como Gerência SNMP. O SNMP é o protocolo de gerência deste padrão. Este padrão descreve não apenas o protocolo de gerência, mas também um conjunto de regras que são usadas para definir as informações de gerência e um conjunto inicial de informações de gerência que já podem ser utilizadas (LOPES; SAUVE; NICOLLETTI, 2003, p. 18).

O uso do SNMP, principal protocolo de gerenciamento de rede, se tornou padrão para diversos dispositivos, saindo da linha de dispositivos tradicionais de rede e incorporando todos dispositivos condizentes com as perspectivas do mercado e que possuem necessidade de estar 100% em funcionamento (BONOMO, 2006).

2.2.1 MIB

A base de informações de gerenciamento, do inglês, *Management Information Base* (MIB), é uma base de dados de objetos gerenciados e rastreados pelos agentes. O MIB é utilizado como uma lista para o agente, com os objetos que ele controla. Um exemplo de objeto armazenado é o “*status*”, onde é possível encontrar dados como “*up, down* ou *testing*”. Esta lista define as informações que o gerente pode utilizar para monitorar o dispositivo no qual o agente está alocado (SOARES, 2013).

Um agente pode implementar muitas MIBs, mas todos os agentes implementam uma MIB particular chamada MIB-II (RFC 1213). Essa norma é responsável por definir variáveis para estatísticas de interface (velocidades de interface, MTU, octetos enviados, octetos recebidos, etc.) e *status* pertencentes ao sistema (localização do sistema, de contato do sistema, etc.). Em suma, o principal objetivo do MIB-II é fornecer a informação TCP/IP geral de gestão (SOARES, 2013).

O protocolo SNMP não define quais informações um sistema deve oferecer, estas informações são definidas pelas MIBs. A MIB, descreve a estrutura de gestão dos dados por meio do uso de denominações hierárquicas de um dispositivo, que contém os identificadores dos objetos, OID (*Object Identifier*). O OID identifica cada variável que pode ser lida ou escrita no dispositivo que o agente está instalado (SOARES, 2013).

2.2.2 Agentes e Gerentes

As aplicações de gerência de redes são baseadas na troca de informações entre um agente e um gerente, neles serão feitas a coleta e gerência dos dados obtidos.

O agente faz a coleta das informações no dispositivo e armazena estas informações na MIB, ele é responsável por fazer o gerenciamento das informações coletadas, atendendo as solicitações enviadas pelo gerente, ou até mesmo envio informações sem a solicitação em caso de falhas, este utiliza a operação “*trap*” para envio de informações ao gerente (BONOMO, 2006).

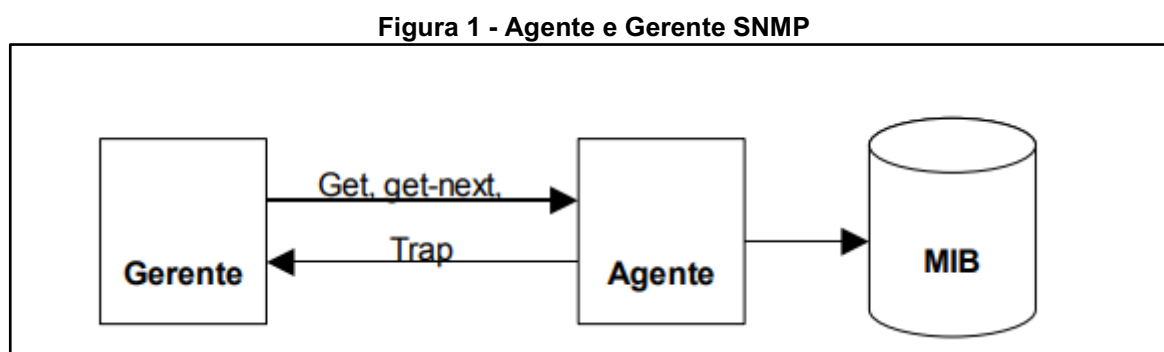
O gerente também conhecido como *Network Management Stations* (NMS), é um servidor que executa um *software* de gerenciamento de rede. O gerente é responsável por coletar informações junto aos agentes, processar essas informações e tomar ações com base nos dados processados, podendo enviar uma ação para o agente executar no dispositivo ou enviar incidentes de alerta ao responsável pela rede, com o objetivo de manter o pleno funcionamento do dispositivo gerenciado (BONOMO, 2006).

2.2.3 Operações SNMP

O SNMP, é um protocolo de requisição e resposta simples, que possui na versão SNMPv1 quatro operações básicas: o *GET* permite que o NMS (*Network Management Stations*) solicite ao agente uma instância de um objeto gerenciado, o *GET-NEXT* permite que o NMS solicite a próxima instância de um objeto gerenciado. Ou seja, se o NMS quiser recuperar todos elementos de uma tabela, ele enviará uma operação “*GET*” e em seguida várias operações “*GET-NEXT*”. Ainda, o *SET* permite

que o NMS possa modificar valores de uma instância de objetos gerenciados pelo agente e o *TRAP* é utilizado pelo agente, permitindo o envio de eventos ao NMS, utilizado em situações de falha (BONOMO, 2006).

A Figura 1 ilustra o esquema da comunicação de um protocolo SNMP com todos os seus elementos.



Fonte: Bonomo (2006)

2.2.4 Versões SNMP

Atualmente o SNMP possui 3 versões. O Quadro 1 apresenta as versões e suas respectivas características.

Quadro 1 - Versões Protocolo SNMP

Versão	Data	Operações Suportadas	Mudanças
V1	1990	get, get-next, set, get-response e trap	Não possui
V2	1996	get-bulk, inform, notification e report	Adicionado novas operações
V3	1999	get-bulk, inform, notification e report	Adicionada maior segurança

Fonte: Bahls (2016)

Atualmente as versões mais utilizadas do SNMP são as versões 1 e 2. A versão 3 ainda não possui grande suporte por partes dos equipamentos de monitoramento (SOARES, 2013). A versão utilizada neste trabalho será o SNMP versão 2.

2.3 SOFTWARE ZABBIX

O Zabbix é uma ferramenta de monitoramento de código aberto, que foi inicialmente desenvolvido por Alexei Vladishev. Atualmente, o desenvolvimento e o suporte da ferramenta são realizados pelo Zabbix SAI (ZABBIX, 2021).

Zabbix é um *software* que monitora vários parâmetros de uma rede e a saúde e integridade de servidores, máquinas virtuais, aplicativos, serviços, bancos de dados, *sites*, nuvem e muito mais. O Zabbix usa um mecanismo de notificação flexível que permite aos usuários configurar alertas baseados em *e-mail* para praticamente qualquer evento. Isso permite uma reação rápida aos problemas do servidor. O Zabbix oferece excelentes recursos de relatório e visualização de dados com base nos dados armazenados. Isso torna o Zabbix ideal para planejamento de capacidade (ZABBIX, 2021).

O Zabbix, suporta as três versões do protocolo SNMP, e possui um agente próprio compatível com diversos sistemas operacionais tais como: Windows, Linux, macOS, AIX, FReeBSD, HP-UX, NetBSD, OpenBSD, SLES, Solaris e Tru64 (ZABBIX, 2021).

2.3.1 Arquitetura Zabbix

O Zabbix é composto por vários componentes de *software* principais, sendo eles Zabbix Server, Zabbix Agent, Zabbix Proxy, Interface Web e Banco de Dados (ZABBIX, 2021).

O Zabbix Server é o componente central. Os agentes enviam para ele as informações e estatísticas com relação a disponibilidade e integridade do equipamento monitorado. Todas as configurações de monitoramento e processamento dos dados recebidos dos equipamentos monitorados, bem como a configuração de tomada de ações e disparo de incidentes são realizadas no Zabbix Server (ZABBIX, 2021).

O Zabbix Agent está presente nos *hosts* monitorados e é responsável pela coleta dos dados e envio ao Zabbix Server ou Zabbix Proxy. Por meio do Agente é possível coletar informações de uso de recursos e monitorar aplicações hospedadas no *host* (ZABBIX, 2021).

O Zabbix Proxy é um componente de coleta. Seu uso é opcional e possui a vantagem de retirar a carga de coleta do Zabbix Server. O Zabbix Proxy se assemelha ao Zabbix Server, porém fará apenas a coleta dos dados de desempenho e disponibilidade, armazenando os dados obtidos e fazendo a transmissão dos dados ao Zabbix Server (ZABBIX, 2021).

A Interface Web permite o administrador da rede interagir com o Zabbix Server, com vantagem de ser acessado de qualquer plataforma. Neste componente é possível fazer as configurações do Zabbix Server e ter acesso às informações coletadas, bem como seus mapas, gráficos e telas (ZABBIX, 2021).

O banco de dados é o componente que irá fazer o armazenamento dos dados de configuração e de dados coletados dos hosts (ZABBIX, 2021).

2.4 GRAFANA

O Grafana é um *software* de código aberto desenvolvido para a análise e visualização de métricas por meio de gráficos. O projeto Grafana foi iniciado por Torkel Ödegaard em 2014, se tornando um dos mais populares no GitHub (GRAFANA, 2021).

O Grafana torna possível a análise de dados por meio de *dashboards*, auxiliando o meio corporativo na tomada de decisões. As principais vantagens do Grafana estão em sua estrutura flexível, tornando a ferramenta de fácil utilização e com possibilidade de integração a diversos bancos de dados e ferramentas por meio de *plugins*. Outra vantagem é sua comunidade participativa, que possibilita a aquisição de *dashboards* pré-configurados (GRAFANA, 2021).

O Grafana consegue consultar diversas fontes de dados, sendo as principais: AWS CloudWatch, Azure Monitor, Elasticsearch, Monitoramento do Google Cloud, Grafite, InfluxDB, Loki, Microsoft SQL Server (MSSQL), MySQL, OpenTSDB, PostgreSQL, Prometheus, Jaeger, Zipkin, Zabbix, Tempo e Dados de teste.

Através do Grafana é possível centralizar os dados em um único local, possibilitando a criação de *dashboards* com dados de diversas fontes alternativas.

2.4.1 Comparando os Dashboards do Zabbix e do Grafana

A Figura 2 apresenta um *dashboard* do Grafana, utilizando dados do Zabbix. Nela é possível verificar a utilização de Memória RAM (*Random Access Memory*) e CPU (*Central Process Unit*), quantidade de *logins* em uma aplicação, requisições em um servidor e a utilização dos discos de armazenamento.

Figura 2 - Dashboard Grafana



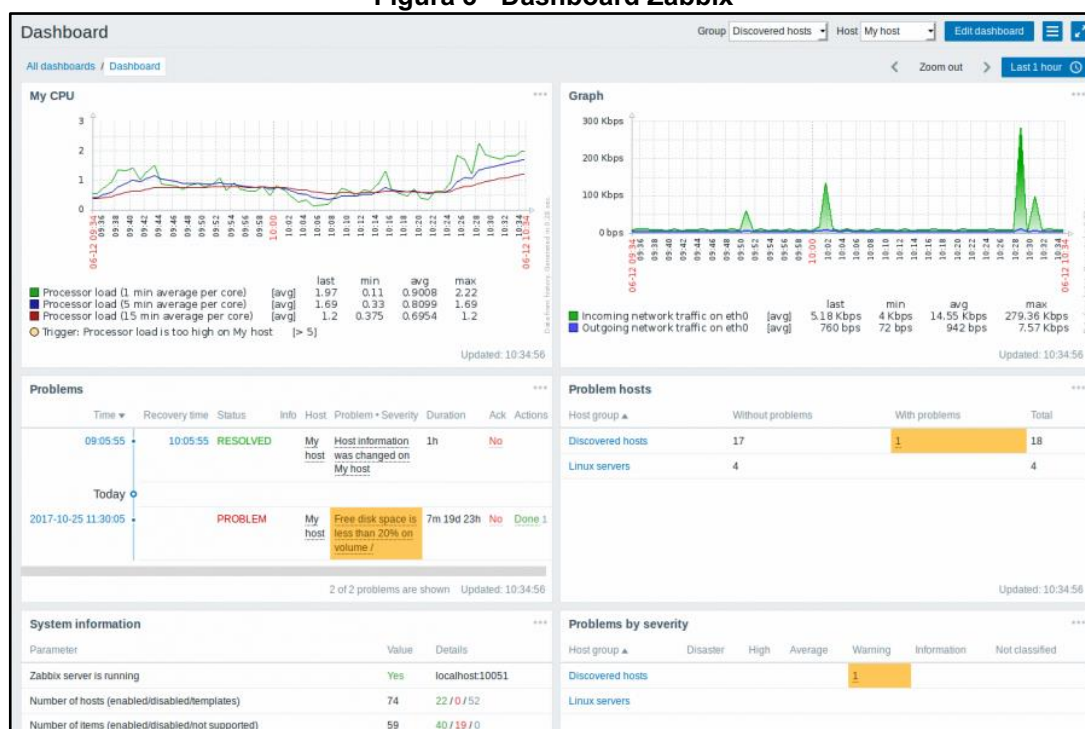
Fonte: Grafana (2021)

Na Figura 3 é possível verificar o modelo de *dashboard* oferecido pela ferramenta Zabbix. Nela são apresentados dados de CPU, utilização de banda de internet, incidentes de servidores monitorados e outras informações referentes aos servidores em monitoramento.

A aplicação Grafana apresenta uma melhor visualização dos *dashboard*, visto que possibilita a personalização de legendas dos gráficos e mesclar com outras fontes de dados. Também é possível verificar que possui sua visualização com paleta de cores que facilita a verificação dos dados, de forma totalmente personalizáveis (GRAFANA, 2021). Já o Zabbix não disponibiliza uma opção para personalizar as cores de cada item dos gráficos e não permite mesclar os dados dos *dashboards* com outras fontes de dados (ZABBIX, 2021).

Outra vantagem apresentada no Grafana é a grande quantidade de tipos de gráficos para a visualização de um mesmo item, possibilitando definir a melhor forma de visualização dos dados (GRAFANA, 2021). No Zabbix os modelos de visualização são predefinidos, tornando a visualização dos dados limitada aos parâmetros definidos pela ferramenta (ZABBIX, 2021).

Figura 3 - Dashboard Zabbix



Fonte: Zabbix (2021)

2.5 TELEGRAM

O Telegram é um serviço de mensagens instantâneas, fundado em 2013 pelos irmãos Nikolai e Pavel Durov, está disponível para smartphones, tablets computadores e aplicações *web* (TELEGRAM, 2021).

O Telegram também é uma ferramenta de código aberto e de utilização gratuita que possui dois tipos de Interface de programação de aplicações (API – *Application Programming Interface*), para integração ao serviço de mensagens.

A API TDLib (biblioteca de banco de dados), possibilita a criação de aplicações Telegram ou clientes personalizados. A API TDLid cuida dos os detalhes

de implementação da rede, criptografia e armazenamento local de dados, tirando estas tarefas do desenvolvedor.

A API de *Bot* (Robô), possibilita a criação de robôs, estes são contas especiais que não requerem número e possibilita a integração com outras ferramentas para envio de mensagens através do protocolo HTTPS (*Hyper Text Transfer Protocol Secure*). Para criação de *bot* no Telegram é necessário possuir acesso a ferramenta e informar no campo de pesquisa o nome “@BotFather”. Ao iniciar a conversa são informadas as instruções para criação do robô (TELEGRAM, 2021).

2.6 TRABALHOS RELACIONADOS

Realizou-se uma pesquisa em trabalhos relacionados ao tema, a fim de encontrar trabalhos que contenham soluções similares a proposta neste projeto. Essa pesquisa resultou em alguns trabalhos dos quais realizou-se uma análise de seus objetivos e resultados obtidos.

O trabalho de conclusão de curso apresentado por Bahls (2016) com tema Monitoramento Proativo do Ambiente de Rede Utilizando a Ferramenta Zabbix apresenta objetivos similares ao trabalho proposto. Alguns objetivos diferem do trabalho proposto, sendo a integração ao Telegram e Grafana e o trabalho de Bahls (2016) possui a finalidade principal de agir sobre incidentes de parada de serviços de forma automática.

Bahls (2016) realizou testes de paradas em serviços nos servidores monitorados a fim de verificar a atuação do Zabbix em relação a correção de incidentes. Com os resultados obtidos através dos testes, o Zabbix apresentou capacidade de restabelecer os serviços em falha imediatamente, ou seja, no mesmo minuto que ocorreu o incidente de parada. Porém Bahls (2016) não implementou formas de alertar o administrador da rede, dificultando o conhecimento de incidentes ocorridos ou resolução de falhas que o Zabbix não foi capaz de solucionar.

Outro trabalho com implementação similar é o trabalho de Fonseca (2018), com título Monitoramento De Ambiente De Redes Utilizado Zabbix. Nele Fonseca (2018) procura analisar a ferramenta Zabbix implementado e verificando os resultados obtidos.

Fonseca (2018) realiza a verificação do ambiente por meio de testes de parada no serviço apache. Também é possível verificar a implementação de alertas similares ao proposto neste trabalho, utilizando a aplicação Telegram.

3 IMPLEMENTAÇÃO DO MONITORAMENTO DE REDE

Este Capítulo apresenta a implementação do monitoramento de rede. A Seção 3.1 apresenta a arquitetura do ambiente de rede que será implementado o monitoramento. Por fim, na Seção 3.2 será apresentado todos os passos para o total monitoramento do ambiente de rede proposto.

3.1 ARQUITETURA DE REDE

A arquitetura da rede utilizada para a realização deste trabalho é apresentada na Figura 4. Ela é formada por 1 servidor Hyperv, 3 servidores virtuais e 1 roteador.

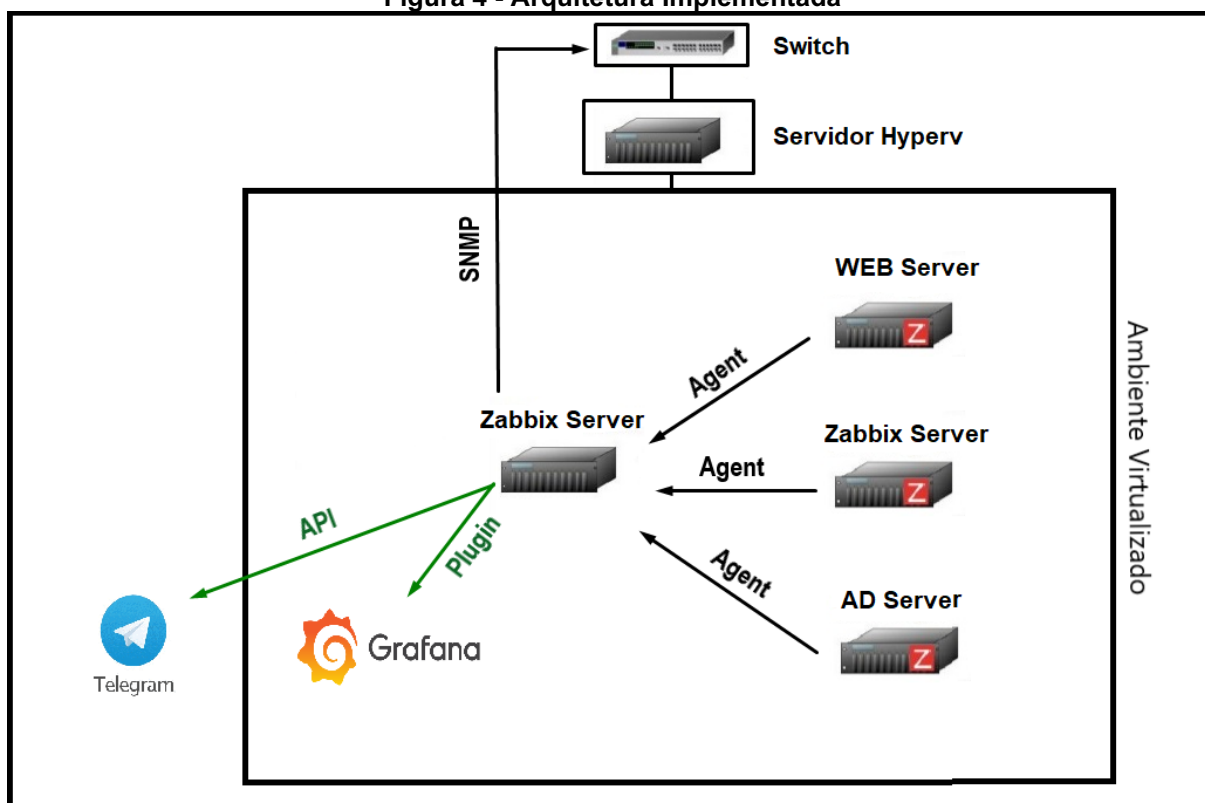
O sistema de monitoramento de rede será implementado em um ambiente virtualizado através de um servidor de virtualização, Hyperv. No ambiente Hyperv estão presentes 3 servidores, sendo o Zabbix Server, AD Server e Web Server.

O AD server é uma VM (*Virtual Machine*), máquina virtual, com SO (Sistema Operacional) Windows Server 2016, nele estão instalados os serviços de *Active Directory* (AD) e *Domain Name System* (DNS). O Web Server é uma VM com SO Linux Debian 10, nele está instalado o serviço Apache. O Zabbix Server é uma VM com SO Linux Debian 10, nele estão instalados todos componentes da ferramenta Zabbix e aplicação Grafana. O Zabbix Server, AD Server e Web Server serão monitorados através do Agent Zabbix, instalados nos servidores.

O ambiente de rede possui um roteador responsável pela comunicação de rede do Servidor Hyperv. O roteador será monitorado no Zabbix através do protocolo SNMP versão 2.

O Telegram será integrado ao Zabbix Server através da API de BOT, já o Grafana, presente no ambiente virtualizado, será integrado através do *plugin* “*alexanderzobnin-zabbix-app*”.

Figura 4 - Arquitetura Implementada



Fonte: Autoria Própria

3.2 ETAPAS DO DESENVOLVIMENTO DO TRABALHO

Para o desenvolvimento deste trabalho, dividiu-se em etapas: 6 etapas, a saber: Etapa 1 - Instalação e Configuração da Ferramenta Zabbix, Etapa 2 - Instalação e Integração do Grafana, Etapa 3 – Integração do Telegram, Etapa 4 – Configuração do Monitoramento de Rede, Etapa 5 – Testes do Monitoramento de Rede, Etapa 6 – Análise dos Testes de Monitoramento de Rede.

3.2.1 Etapa 1 - Instalação e Configuração da Ferramenta Zabbix

A ferramenta Zabbix, versão 5.2 LTS (*Long-Term Support*), foi instalada através dos pacotes presentes no repositório oficial do Zabbix.

O processo de instalação e configuração da ferramenta Zabbix está descrito no Apêndice A.

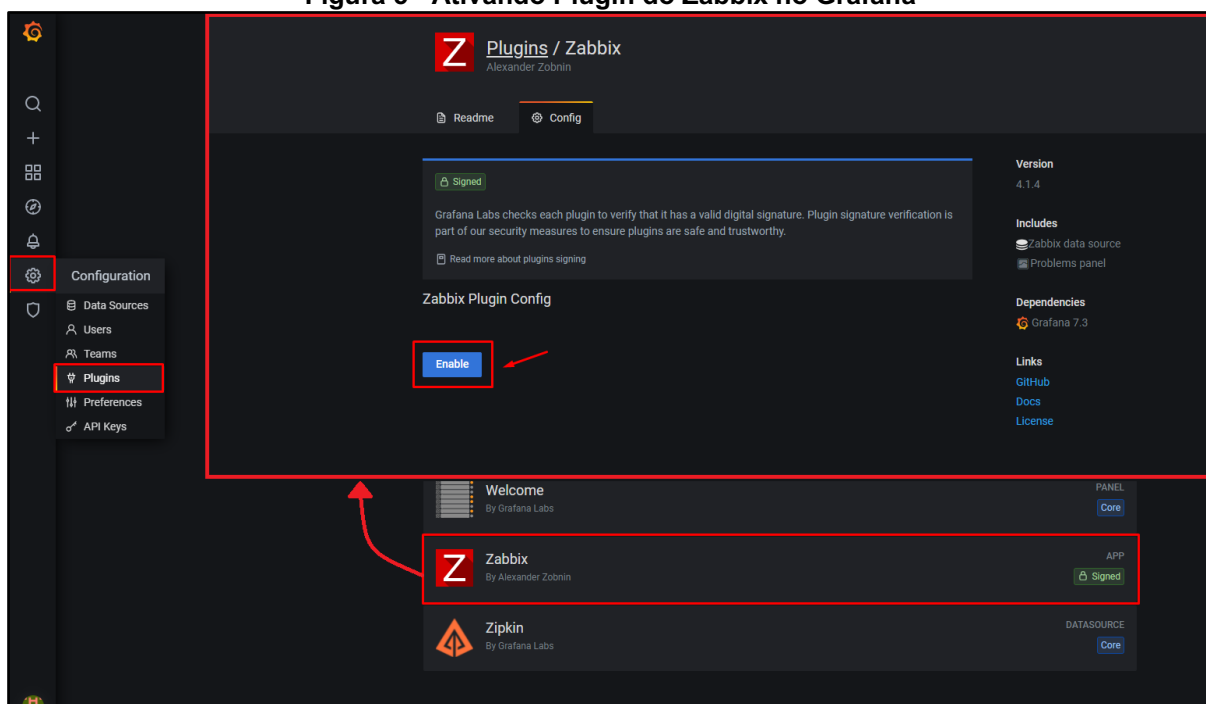
3.2.2 Etapa 2 - Instalação e Integração do Grafana

A aplicação Grafana, versão 7.5.5, foi instalada no Servidor Zabbix com base nos pacotes presentes no repositório oficial do Debian.

Toda a instalação e configuração realizada para o funcionamento da aplicação Grafana, bem como a instalação do *plugin* de integração ao Zabbix, encontra-se no Apêndice A deste documento.

Para integração do Grafana ao Zabbix, faz-se necessário acessar a interface do Grafana. Na opção “*Configuração*→*plugins*→*Zabbix*”, será habilitado o funcionamento do *plugin* de integração ao Zabbix, conforme apresenta a Figura 5.

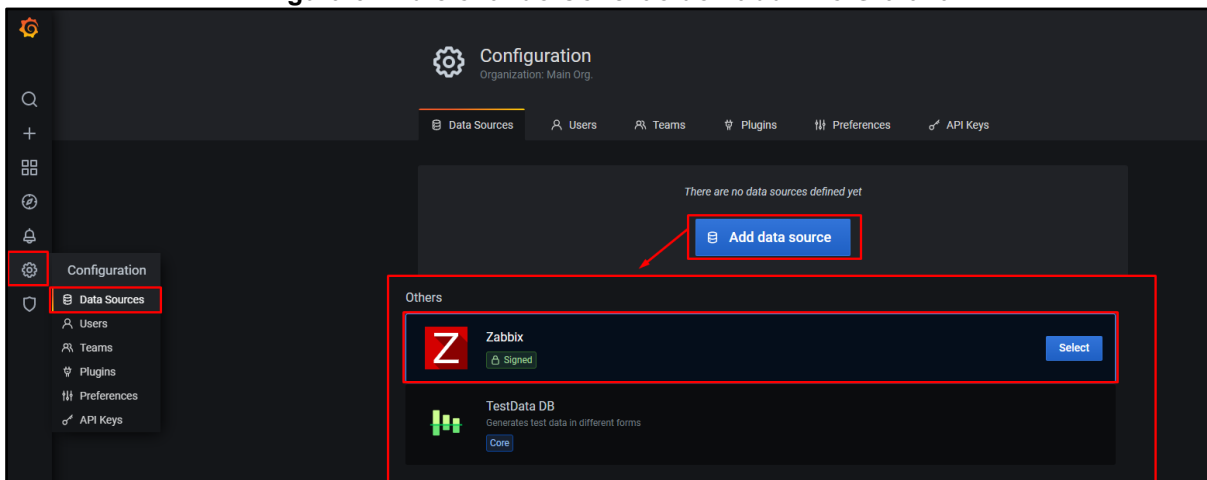
Figura 5 - Ativando Plugin do Zabbix no Grafana



Fonte: Autoria Própria

Ao acessar as opções Zabbix, na aba de *plugins*, foi preciso clicar no botão “*Enable*” para fazer a ativação do *plugin* de integração. Na sequência foi feita a configuração entre o Zabbix e o Grafana, acessando a aba “*Configuração*→*Data Sources*”, ao clicar no botão “*Add data source*” foi selecionada a opção “*Zabbix*”, conforme Remontti (2020), orienta e que ilustra a Figura 6.

Figura 6 - Adicionando Conexão do Zabbix no Grafana



Fonte: Autoria Própria

Para configuração de conexão da ferramenta Zabbix ao Grafana, faz-se necessário acessar a aba “Configurações” na interface web do Grafana. No campo, “HTTP/URL”, foi adicionado o endereço do servidor “http://IP_URL/zabbix/api_jsonrpc.php”, em seguida no campo “HTTP→Acess” selecionou-se a opção “Browser”. No campo “Zabbix API details→Username” foi adicionando um usuário do Zabbix com propriedades de administrador e no campo “Zabbix API details→Password” a senha desse usuário. Por fim clicou-se no botão “Save & Teste”, conforme Remontti (2020), orienta e que ilustra a Figura 7.

Figura 7 - Adicionando Conexão Zabbix no Grafana 2

The image shows the Grafana configuration interface for adding a Zabbix data source. The interface is dark-themed and contains several sections:

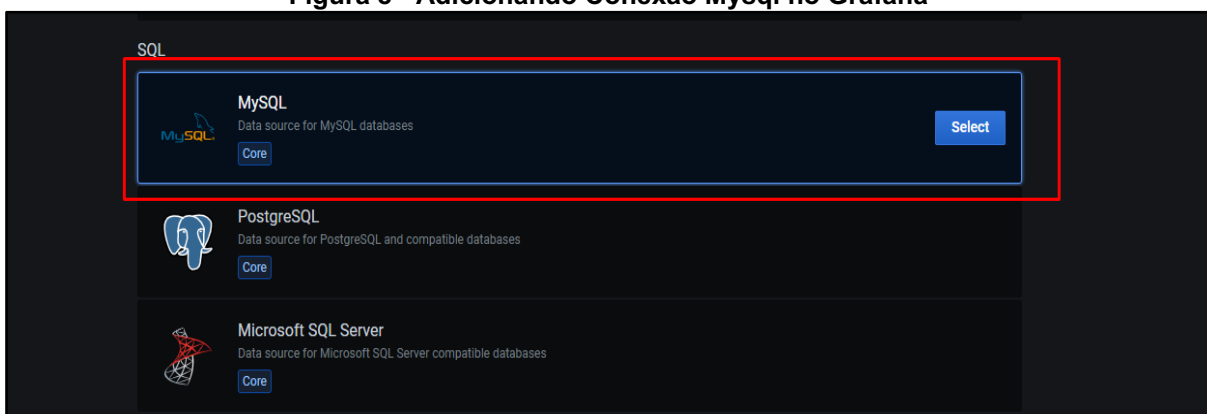
- HTTP**: A red box highlights the URL field containing `http://172.19.7.120/zabbix/api_jsonrpc.php` and the Access dropdown menu set to `Browser`.
- Auth**: Includes a `Basic auth` toggle (disabled) and a `With Credentials` toggle (disabled).
- Zabbix API details**: A red box highlights this section, which includes:
 - `Username`: `Admin`
 - `Password`: `Configured` with a `Reset` button.
 - `Trends`: A toggle switch (disabled).
 - `Cache TTL`: `1h`
 - `Timeout`: `30`
- Direct DB Connection**: Includes an `Enable` toggle (disabled).
- Other**: Includes `Disable acknowledges for read-only users` and `Disable data alignment` toggles (both disabled).

A green success message at the bottom states: `✓ Zabbix API version: 5.0.11`. At the bottom left, a red arrow points to a red box around the `Save & Test` button, with `Delete` and `Back` buttons also visible.

Fonte: Autoria Própria

O Grafana pode utilizar um servidor de banco de dados para armazenar as consultas. Com isso é possível armazenar os itens de consultas para dar velocidade ao processo de exibição e configuração. Para configurar o banco foi pré-instalado e configurado um servidor de banco de dados. A conexão foi adicionada na aba “Configuração→Data sources”, clicando no botão “Add data source” e selecionando a opção “MySQL”, conforme Remontti (2020), orienta e que ilustra a Figura 8.

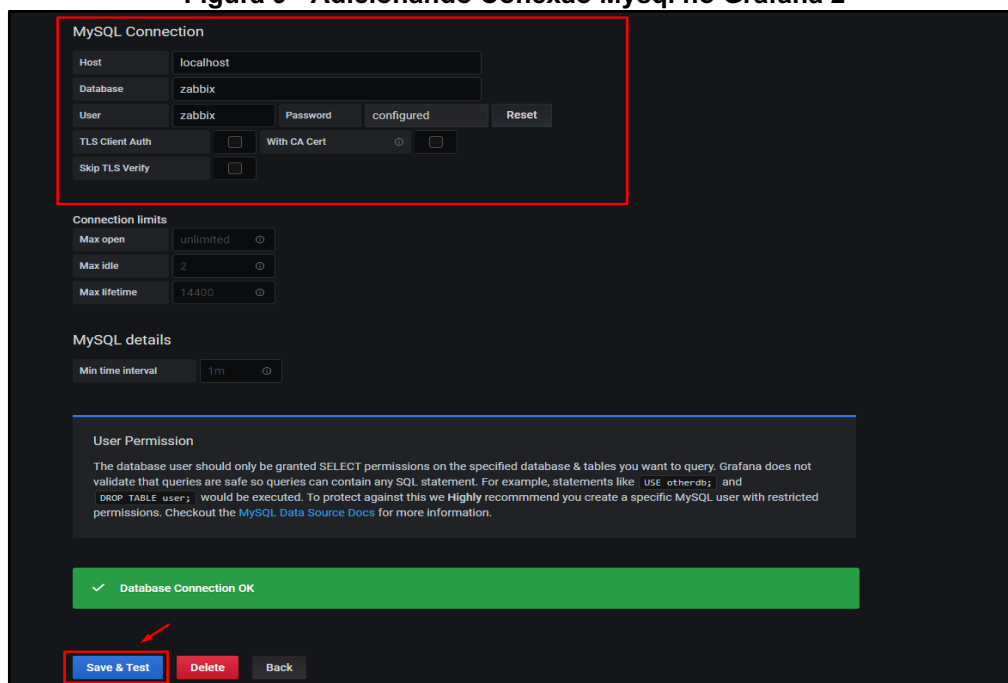
Figura 8 - Adicionando Conexão Mysql no Grafana



Fonte: Autoria Própria

A Figura 9 mostra a configuração do banco. Foi informado no campo “*Host*”, endereço do servidor do banco de dados, ou seja, é ele que permite a conexão com o banco, no campo “*Database*”, o nome da base de dados do Zabbix, no campo “*User*”, o usuário de acesso ao banco e no campo “*Password*” a senha do usuário de acesso ao banco. Por fim, clicou-se no botão “*Save & Teste*” para concluir e testar a configuração (REMONTTI, 2020).

Figura 9 - Adicionando Conexão Mysql no Grafana 2

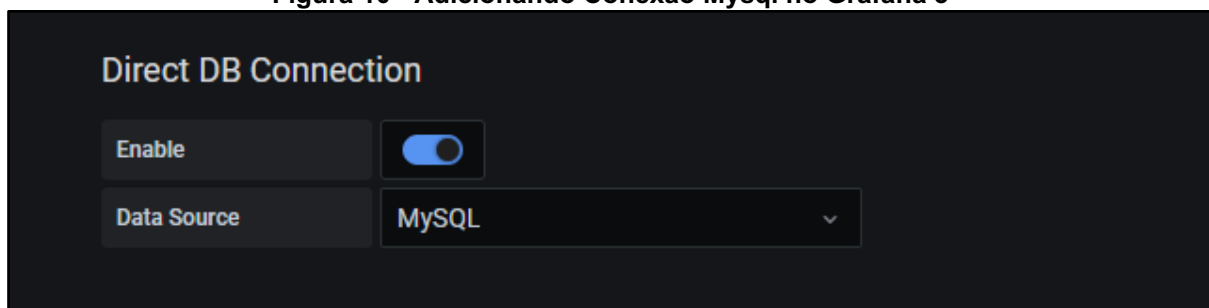


Fonte: Autoria Própria

Após a configuração do Banco de dados foi necessário voltar na configuração de conexão com o Zabbix e habilitar o “*Direct DB connection*”, conforme apresenta a

Figura 10, para então informar o tipo de banco de dados que foi utilizando (REMONTTI, 2020).

Figura 10 - Adicionando Conexão Mysql no Grafana 3

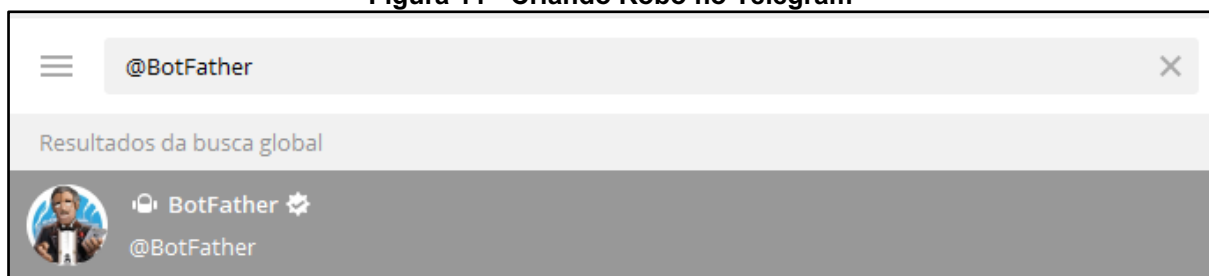


Fonte: Autoria Própria

3.2.3 Etapa 3 – Integração do Telegram

A integração do Zabbix ao Telegram foi feita através de um robô, Bot, que simula ações humanas. O robô do Telegram é o responsável por coletar as informações enviadas do Zabbix. O primeiro passo para a integração das plataformas foi a criação de um *Bot* no Telegram, acessando a plataforma e informando no campo de pesquisa de usuário o comando “@BotFather”, conforme exibido na Figura 11 (REMONTTI, 2020).

Figura 11 - Criando Robô no Telegram

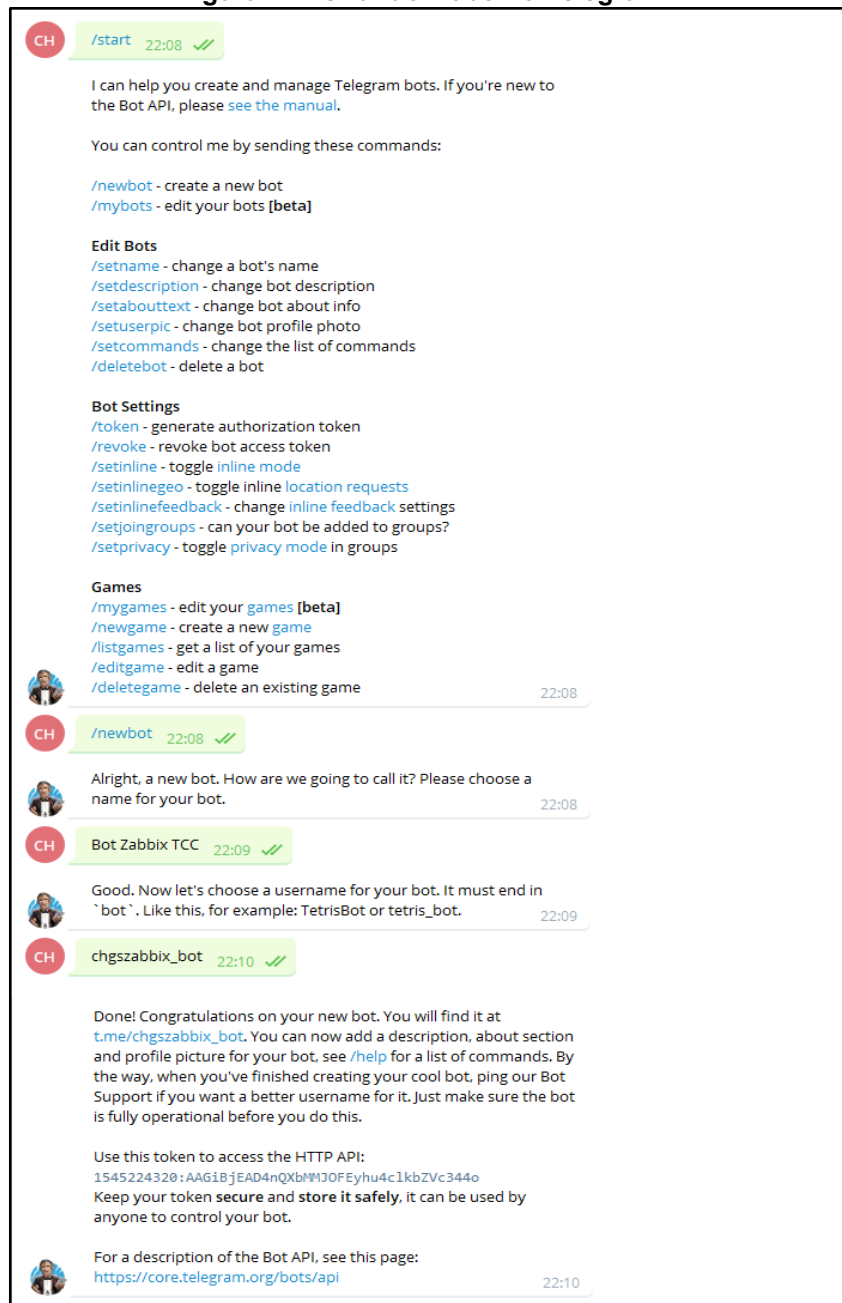


Fonte: Autoria Própria

O próximo passo, foi iniciar uma conversa clicando no robô. O Telegram dispara automaticamente o comando “/start” ao entrar na conversa, então, apresenta uma lista com todos os comandos disponíveis. O comando “/newbot”, possibilita a criação de um nome para o robô recém-criado. Na sequência, foi configurado um *username* para a robô, ou seja, o nome de acesso ao *bot*, seguindo o padrão “*nome_bot*”. O *Bot* foi criado seguindo os passos acima e seu acesso direto

disponibilizado bem como o *token* de acesso a API HTTP (*Hypertext Transfer Protocol*), conforme apresentado na Figura 12 (REMONTTI, 2020).

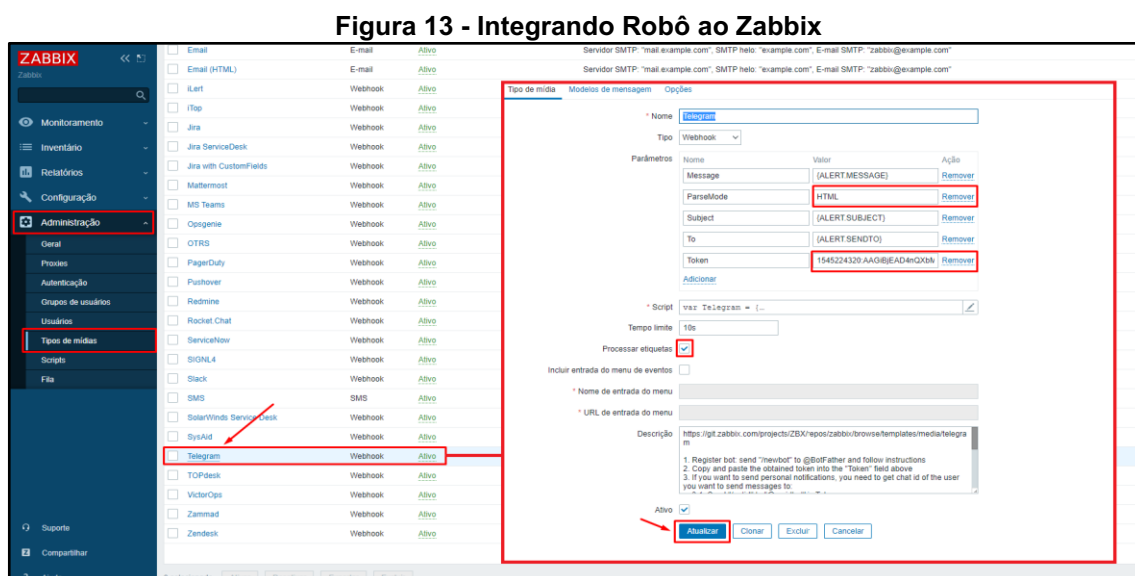
Figura 12 - Criando Robô no Telegram 2



Fonte: Autoria Própria

A integração do robô recém-criado ao Zabbix, foi feita por meio da interface do Zabbix. Na aba, “*Administração*→*Tipos de mídias*”, ilustrada pela Figura 13 é possível observar a opção relacionada ao Telegram. Ainda na Figura 13, na aba, “*Telegram*→*Tipos de mídia*” foram preenchidos os campos “*PaseMode*” com “*HTML*”, e o campo “*Token*” com o *token* do robô, também foi selecionado o *check* do campo

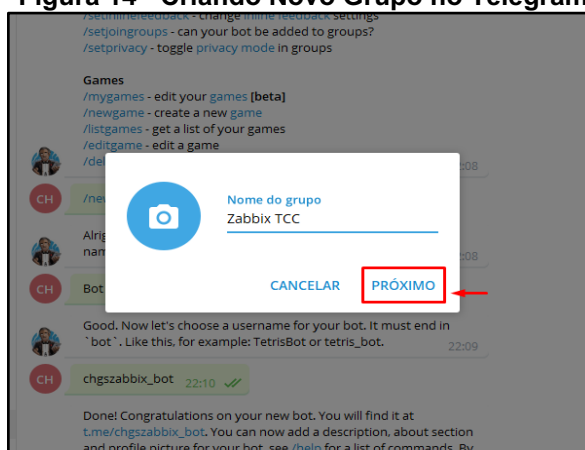
“Processas etiquetas”. E para finalizar clicou-se no botão “Adicionar” (REMONTTI, 2020).



Fonte: Autoria Própria

Prosseguindo com a configuração do Telegram, foi feita a criação de um grupo, para receber os alertas do Zabbix, selecionando a opção “Novo Grupo”. A Figura 14, mostra que, após clicar na referida opção foi solicitado o preenchimento do nome do grupo e na sequência a adição de um novo usuário ao grupo (REMONTTI, 2020).

Figura 14 - Criando Novo Grupo no Telegram

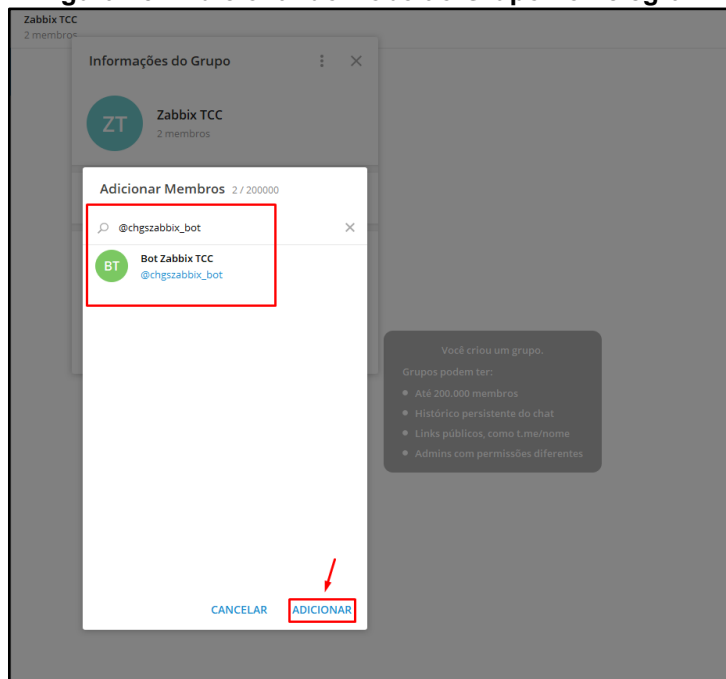


Fonte: Autoria Própria

A etapa seguinte consistiu-se em adicionar o *Bot* ao grupo criado, na aba de configuração do grupo em “Adicionar Membros”, informando o *username* do *Bot* e

clicando em “Adicionar” para finalizar, conforme o ilustrado pela Figura 15 (REMONTTI, 2020).

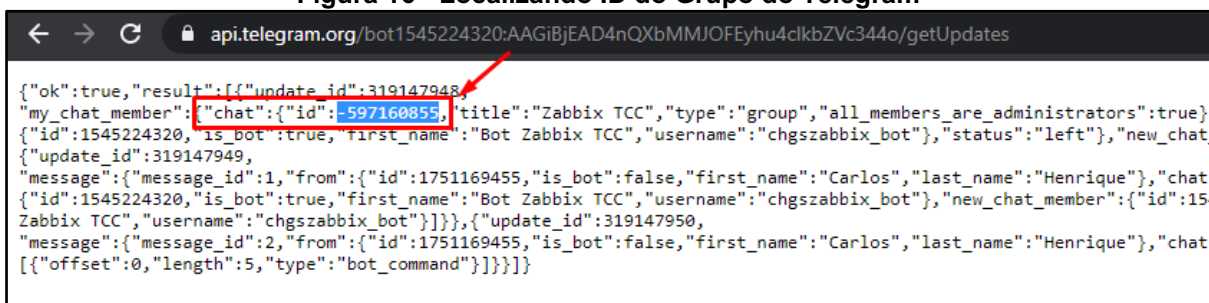
Figura 15 - Adicionando Robô ao Grupo no Telegram



Fonte: Autoria Própria

Dando continuidade à integração do Telegram com o Zabbix, foi preciso digitar um texto aleatório no grupo, para então, conseguir acessar a API do Zabbix pelo *link* “<https://api.telegram.org/botTOKEN/getUpdates>”. Nesse passo, foi feita a substituição do “*TOKEN*”, pelo *token* recebido durante a criação do *Bot*. A Figura 16, mostra a localização o ID (*Identification*) do chat do grupo após o acesso à API do Telegram (REMONTTI, 2020).

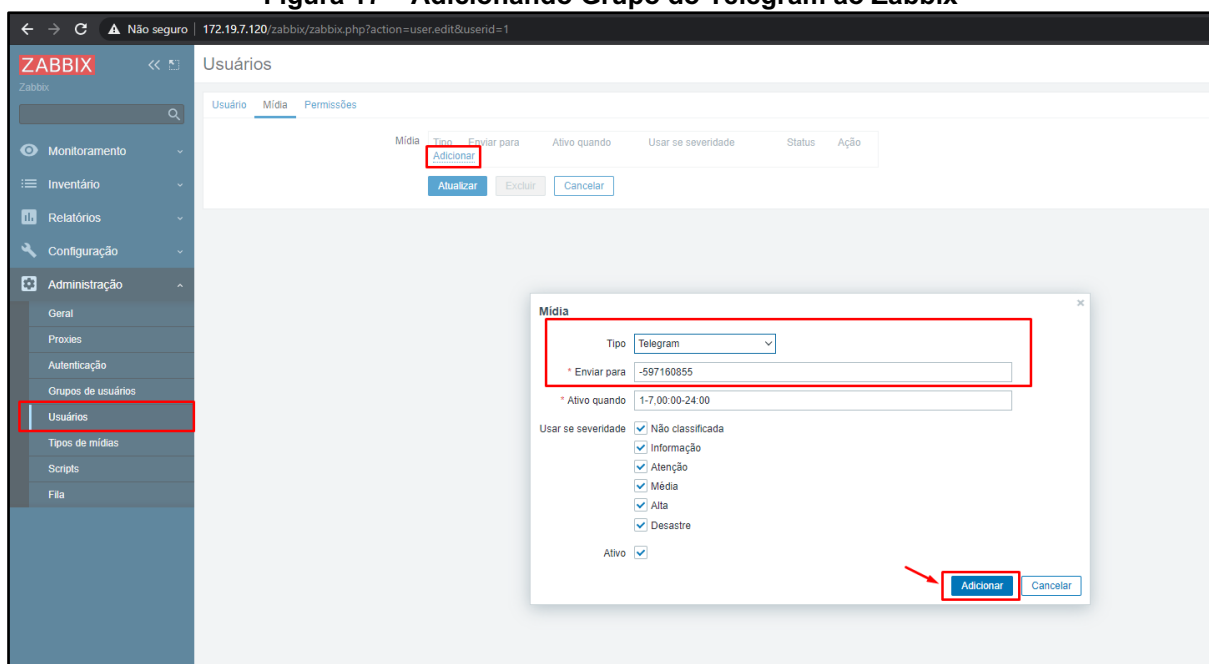
Figura 16 - Localizando ID do Grupo do Telegram



Fonte: Autoria Própria

Agora, dentro da interface do Zabbix, foi selecionada a opção “Administração→Usuários” para selecionar o usuário “admin” e acessar a aba “Mídia”. Para então, selecionar no campo “Tipo” a opção “Telegram” e no campo “Enviar para” o ID do grupo obtido na API do Telegram, descrito anteriormente e ilustrado pela Figura 16. Já na Figura 17, é possível ver essa inclusão do grupo do Telegram ao Zabbix (REMONTTI, 2020).

Figura 17 – Adicionando Grupo do Telegram ao Zabbix



Fonte: Autoria Própria

Neste ponto o Telegram foi completamente integrado ao Zabbix, porém, para que os diferentes alertas fossem enviados ao Telegram foi preciso criar ações de envio. Para programar as ações acessou-se a opção “Configuração→Ações” e clicando no botão “Criar ação”. Na aba, “Ação” foi descrito o nome da ação, e na opção “Adicionar” do campo “Condições” dentro a aba “Nova condição”, foi selecionado a opção “Incidente suprimido” no campo “Tipo” e a opção “Não” no campo “Operador” conforme mostra a Figura 18, por fim clicou-se em “Adicionar” (REMONTTI, 2020).

Figura 18 - Criando Nova Ação de Envio Para o Telegram

The screenshot shows the Zabbix 'Ações' configuration page. The 'Operações' tab is selected. A new action named 'Notificar Via Telegram' is being configured. The 'Condições' section is highlighted with a red box, and a modal window titled 'Nova condição' is open. In the modal, the 'Tipo' dropdown is set to 'Incidente suprimido' and the 'Operador' is set to 'Não'. The 'Adicionar' button in the modal is also highlighted with a red box.

Fonte: Autoria Própria

Na próxima etapa foram criadas as operações de envio para o Telegram, na aba “Ações” a sub aba “Operações”. O cadastro das operações de problemas do Zabbix foi realizado clicando no botão “Adicionar” do campo “Operações”, onde, dentro da aba “Detalhes da operação”, foi adicionado o usuário “Admin” no campo “Send to users” e no campo “Enviar apenas para” foi selecionada a opção “Telegram”, no campo “Mensagem personalizada” clicou-se para marcar o *checkbox*. Ainda, na aba foi informado no campo “Assunto” o comando **X Problema: {HOST.NAME}**, e no campo “Mensagem” o seguinte comando (REMONTTI, 2020):

```
{EVENT.NAME}
<b>{ITEM.NAME1}</b> <i>{ITEM.VALUE1}</i>

<a href="{HOST.IP}">{HOST.IP}</a>
<i>{EVENT.SEVERITY}</i>
```

Após o correto preenchimento de todos os campos da operação de problema, clicou-se no botão “Add”, conforme a Figura 19 ilustra (REMONTTI, 2020).

Figura 19 - Ação de Envio Para o Telegram - Operações

The screenshot shows the Zabbix Actions configuration interface. The 'Operações' tab is selected, and the 'Adicionar' button is highlighted. The 'Detalhes da operação' dialog box is open, showing the following configuration:

- Tipo da operação:** Enviar mensagem
- Passos:** 1 - 1 (0 - infinitamente)
- Duração do passo:** 0 (0 - usar a ação padrão)
- Send to user groups:** Grupo de usuários (Adicionar)
- Send to users:** Usuário (Admin (Zabbix Administrator) (Adicionar))
- Enviar apenas para:** Telegram
- Mensagem personalizada:**
- Assunto:** Problema: {HOST.NAME}
- Mensagem:**

```
[EVENT.NAME]
<b>{ITEM.NAME1}</b> <i>{ITEM.VALUE1}</i>
<a href="{HOST.IP}">{HOST.IP}</a>
<i>{EVENT.SEVERITY}</i>
```
- Condições:** Texto (Adicionar)

The 'Add' button is highlighted with a red arrow.

Fonte: Autoria Própria

Para o cadastro das operações de recuperação do Zabbix clicou-se no botão “Adicionar” do campo “Operações de recuperação”, e na aba “Detalhes da operação” foi adicionado o usuário “Admin” no campo “Send to users”, no campo “Enviar apenas para” selecionou-se a opção “Telegram” e no campo “Mensagem personalizada” foi marcado o *checkbox*. Ainda na aba “Detalhes da operação” foi informado no campo “Assunto” o comando “**✓ Resolvido: {HOST.NAME}**” e na aba “Detalhes da operação” no campo “Mensagem” o seguinte comando (REMONTTI, 2020):

```
{EVENT.NAME}
<b>{ITEM.NAME1}</b> <i>{ITEM.VALUE1}</i>

<a href="{HOST.IP}">{HOST.IP}</a>
<i>{EVENT.SEVERITY}</i>
```

Na Figura 20, é possível observar o preenchimento de todos os campos da operação de recuperação concluindo essa etapa ao clicar no botão 'Add' (REMONTTI, 2020).

Figura 20 - Ação de Envio Para o Telegram - Recuperação

The screenshot shows the Zabbix Actions configuration interface. A modal dialog titled "Detalhes da operação" is open, showing the configuration for sending a message via Telegram. The dialog includes the following fields and options:

- Tipo da operação:** Enviar mensagem
- Send to user groups:** Grupo de usuários (Adicionar)
- Send to users:** Admin (Zabbix Administrator) (Adicionar)
- Enviar apenas para:** Telegram
- Mensagem personalizada:**
- Assunto:** Resolvido: {HOST.NAME}
- Mensagem:** {EVENT.NAME}
{ITEM.NAME 1} <i>{ITEM.VALUE 1}</i>
{HOST.IP}
<i>{EVENT.SEVERITY}</i>

The "Add" button is highlighted with a red arrow, indicating the final step in the configuration process.

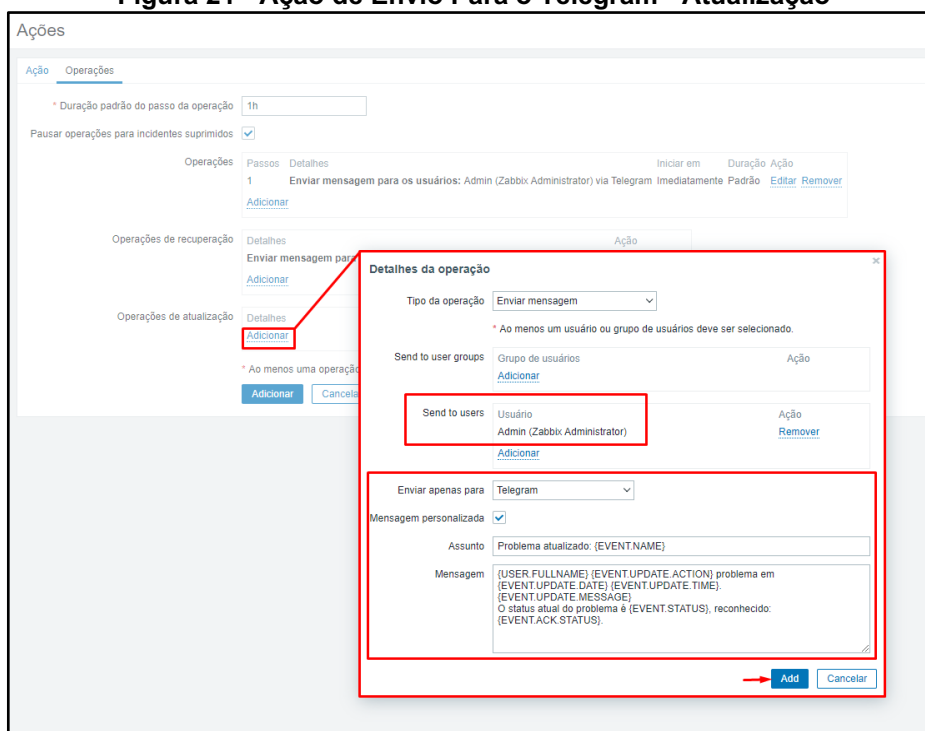
Fonte: Autoria Própria

Para o cadastro das operações de atualização do Zabbix clicou-se no botão "Adicionar" no campo "Operações de atualização", onde foi aberta a aba "Detalhes da operação". Na aba, "Detalhes da operação" foi adicionado o usuário "Admin" no campo "Send to users", no campo "Enviar apenas para" foi selecionada a opção "Telegram", no campo "Mensagem personalizada" foi apenas selecionado o checkbox e no campo "Assunto" o comando "Problema atualizado: {EVENT.NAME}". Ainda, na aba "Detalhes da operação" foi informado o seguinte comando no campo "Mensagem" (REMONTTI, 2020):

```
{USER.FULLNAME} {EVENT.UPDATE.ACTION} problema em {EVENT.UPDATE.DATE}
{EVENT.UPDATE.TIME}.
{EVENT.UPDATE.MESSAGE}
O status atual do problema é {EVENT.STATUS}, reconhecido: {EVENT.ACK.STATUS}.
```

Na Figura 21 é ilustrada a finalização deste item que ocorre após o preenchimento de todos os campos na operação de atualização e concluindo a etapa clicando no botão “Add” (REMONTTI, 2020).

Figura 21 - Ação de Envio Para o Telegram - Atualização



Fonte: Autoria Própria

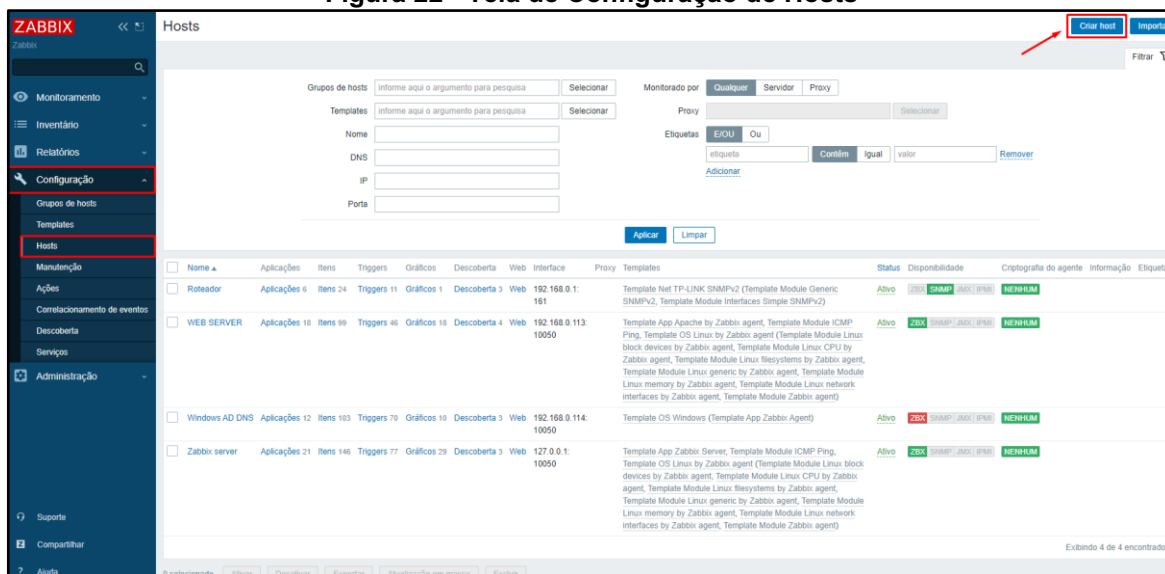
Após concluir a configuração de todas as operações, finaliza-se a ação de envio de operações clicando no botão “Adicionar”.

3.2.4 Etapa 4 – Configuração do Monitoramento de Rede

Os *templates* predefinidos que coletam informações dos servidores e exibem o monitoramento foram obtidos por meio do link “<https://share.zabbix.com/official-templates>” (ZABBIX, 2021).

Os hosts foram incluídos através da interface do Zabbix na opção “Configuração→Hosts”, no botão “Criar host”, abrindo essa tela, são exibidos os hosts já adicionados ao Zabbix, conforme mostra a Figura 22. A inclusão dos *hosts* garante o início do monitoramento dos servidores, ou seja, sendo estes quaisquer dispositivos conectados à rede com um endereço IP (*Internet Protocol*) (ZABBIX, 2021).

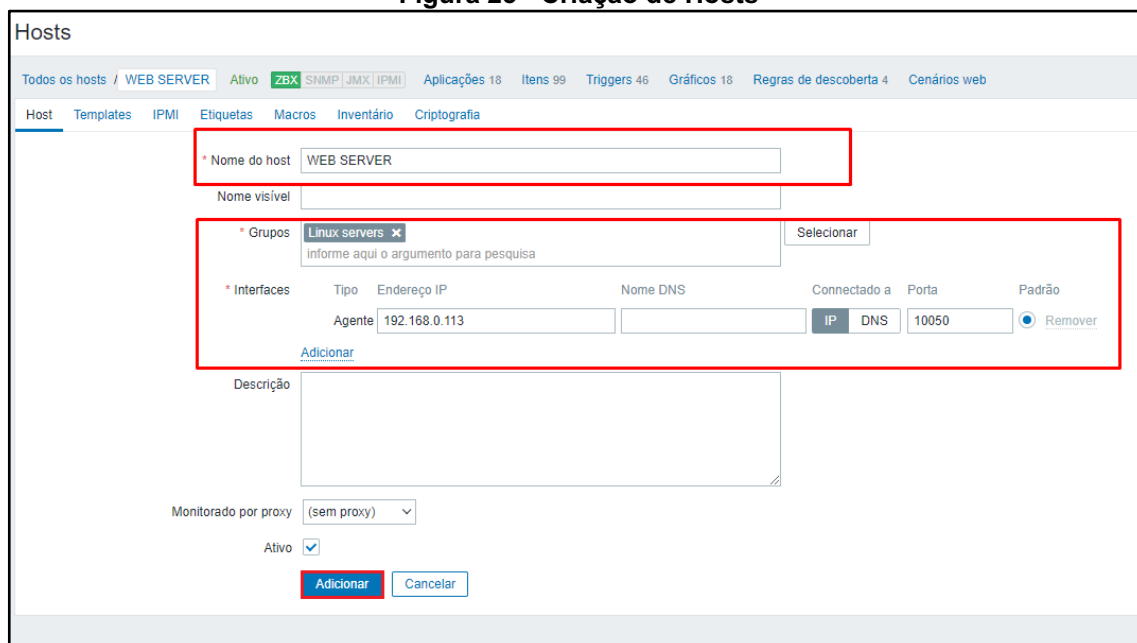
Figura 22 - Tela de Configuração de Hosts



Fonte: Autoria Própria

A Figura 23 ilustra a configuração do *host*, ao clicar no botão “Criar Host” a aba “Hosts” é aberta e aparecem os campos de configuração. No campo “Nome do host” foi informado o nome do servidor que será monitorado, no campo “Grupos”, o grupo que o servidor participa e no campo “Interfaces” o endereço IP do servidor, por último clicou-se no botão adicionar.

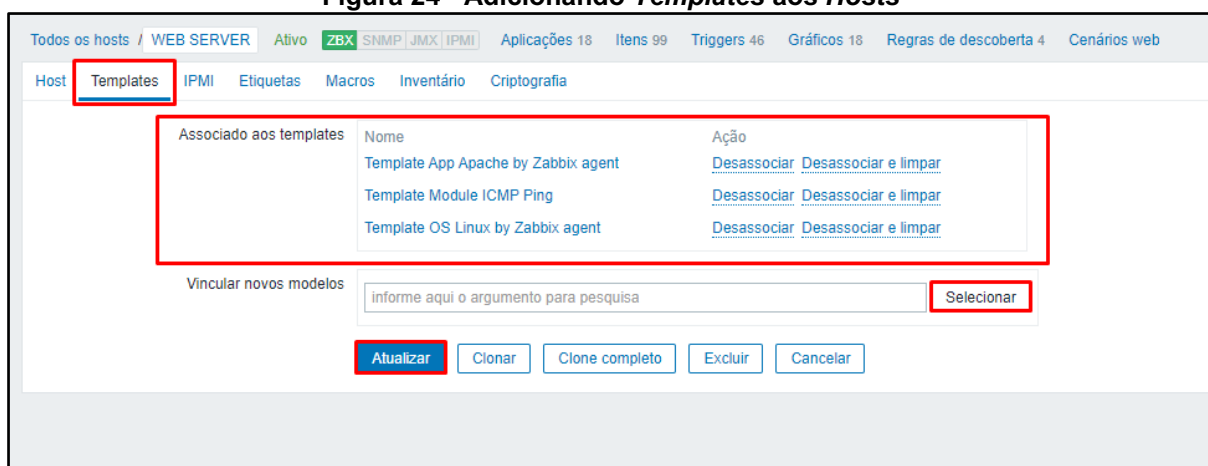
Figura 23 - Criação de Hosts



Fonte: Autoria Própria

Por padrão, o Zabbix já possui alguns grupos configurados, no entanto, a criação de novos grupos pode ser feita na opção “*Configuração→Grupos de hosts*”. Para o monitoramento do host, foi necessário selecionar os *templates* indicados para o determinado tipo de *host*, durante o cadastro do *host* dentro da aba “*Templates*”, após a adição dos *templates* clicou-se no botão “*Atualizar*”, apresentado na Figura 24.

Figura 24 - Adicionando *Templates* aos *Hosts*



Fonte: Autoria Própria

3.2.5 Etapa 5 – Testes do Monitoramento de Rede

A etapa de testes, constitui-se dos testes de monitoramento do Zabbix, utilizando a aplicação Telegram e Grafana para exibir os resultados. Nesta etapa, foi realizado a parada e início do serviço Apache no servidor Webserver, e verificado a capacidade do Zabbix em informar o monitoramento em tempo real. A etapa de testes do monitoramento reativo utilizando o Telegram é apresentada no capítulo 4.2.

No Grafana foram realizados testes de monitoramento em tempo real, apresentando detalhes de recursos dos servidores monitorados. Os testes com a aplicação Grafana encontram-se no capítulo 4.3.

3.2.6 Etapa 6 – Análise dos Testes de Monitoramento de Rede.

A análise dos testes de monitoramento de rede é descrita no capítulo 4.4. Nele são descritos os dados obtidos com os testes e suas características, bem como a comparação com resultados obtidos com trabalhos relacionados.

4 TESTES E RESULTADOS

Este Capítulo apresenta os testes e resultados obtidos através do desenvolvimento do trabalho. A Seção 4.1 apresenta o ambiente configurado que será testado nas seções seguintes. A Seção 4.2 expõe os testes realizados no monitoramento do Zabbix integrado ao Telegram. A Seção 4.3 exhibe os testes realizados no monitoramento do Zabbix integrado ao Grafana. Por fim, na Seção 4.4 são analisados os testes e comparado com dados obtidos por outros autores.

4.1 DESCRIÇÃO DO AMBIENTE TESTADO

O Quadro 2, apresenta o resumo dos servidores e equipamentos de rede monitorados, juntamente com informações do sistema operacional ou modelo do equipamento monitorado e seus respectivos endereços IP.

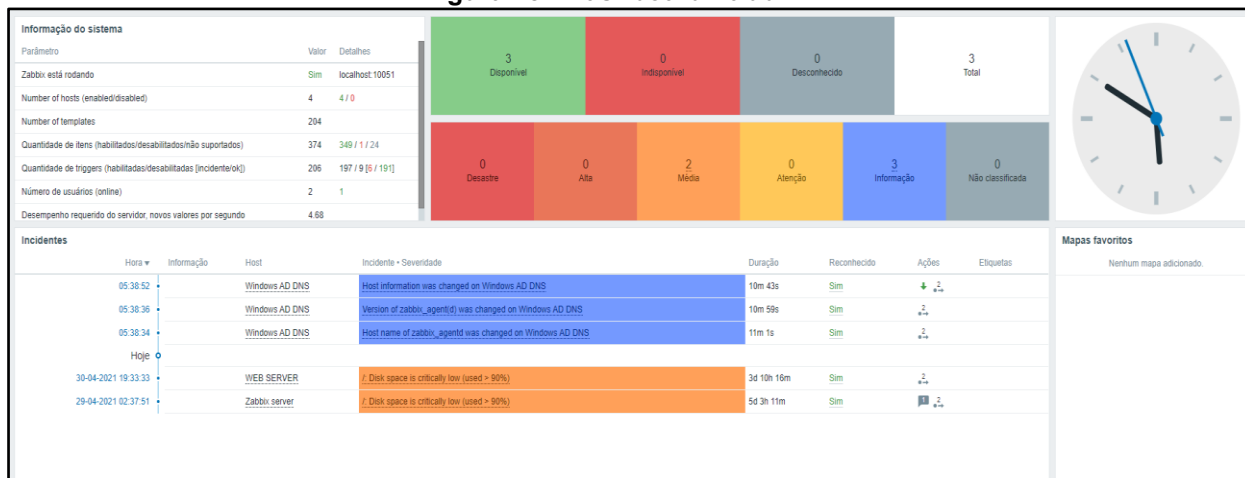
Quadro 2 - Itens Monitorados Pelo Zabbix

Nome	SO / Modelo	Endereço IP
Zabbix	Debian 10	192.168.0.109
Web Server	Debian 10	192.168.0.113
AD_DNS	Windows Server 2016	192.168.0.114
Roteador	TP Link	192.168.0.1

Fonte: Autoria Própria

A interface Web do Zabbix permite que o usuário acompanhe os incidentes ocorridos de forma visual. A Figura 25 apresenta um resumo da estrutura monitorada pelo Zabbix, na parte superior é possível verificar os dados referentes aos servidores ativos e monitorados, bem como as configurações referentes a estes servidores. Ainda na Figura 25, na parte inferior são apresentados os incidentes ocorridos nos servidores monitorados, sendo elas de atualização de dados e alerta de disco próximo a capacidade máxima de armazenamento.

Figura 25 - Dashboard Zabbix



Fonte: Autoria Própria

Na Figura 26 observa-se a tela de configuração dos hosts. Nela é apresentado um resumo dos hosts monitorados, com seus nomes e os itens que estão sendo monitorados. Também é possível notar os *templates* utilizados nos monitoramentos dos *hosts* e os status e forma de monitoramento de cada item.

Figura 26 - Interface dos Hosts Monitorados

Nome	Aplicações	Itens	Triggers	Gráficos	Descoberta	Web	Interface	Proxy	Templates	Status	Disponibilidade	Criptografia do agente	Informação
Roleador	Aplicações 6	Itens 24	Triggers 11	Gráficos 1	Descoberta 3	Web	192.168.0.1:161		Template Net TP-LINK SNMPV2 (Template Module Generic SNMPV2, Template Module Interfaces Simple SNMPV2)	Ativo	ZBX SNMP JMX IPMI	NENHUM	
WEB SERVER	Aplicações 18	Itens 99	Triggers 46	Gráficos 18	Descoberta 4	Web	192.168.0.113:10050		Template App Apache by Zabbix agent, Template Module ICMP Ping, Template OS Linux by Zabbix agent (Template Module Linux block devices by Zabbix agent, Template Module Linux CPU by Zabbix agent, Template Module Linux filesystems by Zabbix agent, Template Module Linux generic by Zabbix agent, Template Module Linux memory by Zabbix agent, Template Module Linux network interfaces by Zabbix agent, Template Module Zabbix agent)	Ativo	ZBX SNMP JMX IPMI	NENHUM	
Windows AD DNS	Aplicações 12	Itens 103	Triggers 70	Gráficos 10	Descoberta 3	Web	192.168.0.114:10050		Template OS Windows (Template App Zabbix Agent)	Ativo	ZBX SNMP JMX IPMI	NENHUM	
Zabbix server	Aplicações 21	Itens 146	Triggers 77	Gráficos 29	Descoberta 3	Web	127.0.0.1:10050		Template App Zabbix Server, Template Module ICMP Ping, Template OS Linux by Zabbix agent (Template Module Linux block devices by Zabbix agent, Template Module Linux CPU by Zabbix agent, Template Module Linux filesystems by Zabbix agent, Template Module Linux generic by Zabbix agent, Template Module Linux memory by Zabbix agent, Template Module Linux network interfaces by Zabbix agent, Template Module Zabbix agent)	Ativo	ZBX SNMP JMX IPMI	NENHUM	

Fonte: Autoria Própria

4.2 TESTES ZABBIX E TELEGRAM

Para os testes o servidor Web Server foi utilizado a fim de simular a parada do serviço Apache. No terminal do servidor, foi executado o comando “*systemctl stop apache2*”, que faz a parada do serviço e na Figura 27 é possível observar o horário de parada do serviço as 05:29.

Figura 27 - Teste de Parada Serviço Apache

```

root@WEBSERVER:~# systemctl stop apache2
root@WEBSERVER:~# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset:
   Active: inactive (dead) since Tue 2021-05-04 05:29:32 -03; 3s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 7420 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Process: 7603 ExecStop=/usr/sbin/apachectl stop (code=exited, status=0/SUCCESS)
   Main PID: 7424 (code=exited, status=0/SUCCESS)

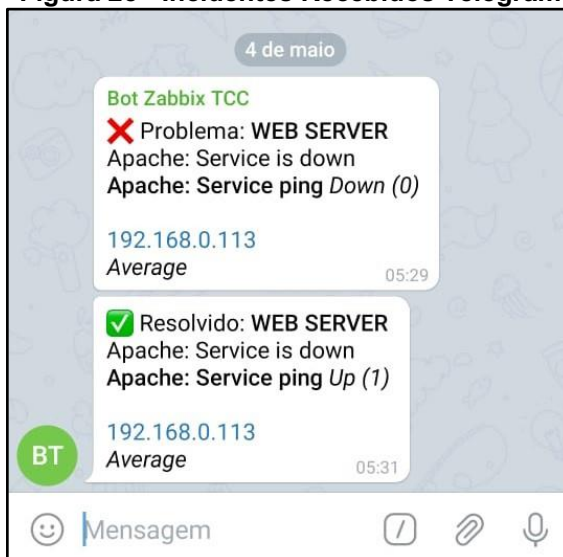
mai 04 05:23:06 WEBSERVER systemd[1]: Starting The Apache HTTP Server...
mai 04 05:23:06 WEBSERVER apachectl[7420]: AH00558: apache2: Could not reliably
mai 04 05:23:06 WEBSERVER systemd[1]: Started The Apache HTTP Server.
mai 04 05:29:32 WEBSERVER systemd[1]: Stopping The Apache HTTP Server...
mai 04 05:29:32 WEBSERVER apachectl[7603]: AH00558: apache2: Could not reliably
mai 04 05:29:32 WEBSERVER systemd[1]: apache2.service: Succeeded.
mai 04 05:29:32 WEBSERVER systemd[1]: Stopped The Apache HTTP Server.
lines 1-15/15 (END)

```

Fonte: Autoria Própria

Na Figura 28 verifica-se o alerta enviado ao Telegram notificando que o serviço apache está parado, *down*, nota-se que o envio da notificação do incidente está com o mesmo horário de parada do serviço apache, 05:29, apresentado na Figura 27.

Figura 28 - Incidentes Recebidos Telegram



Fonte: Autoria Própria

Após efetuar a parada do serviço apache, o comando “*systemctl start apache2*” faz o início do serviço. Na Figura 29, verifica-se o horário de início, as 05:31, e os comandos executados. Após efetuar o início do serviço, o Zabbix envia uma notificação ao Telegram, informando que o problema de parada do Apache foi

resolvido. Na Figura 28 é possível verificar que o horário de notificação de resolução do problema é o mesmo que o comando de início do serviço Apache.

Figura 29 – Teste de Início Serviço Apache

```

root@WEBSERVER:~# systemctl start apache2
root@WEBSERVER:~# systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset:
   Active: active (running) since Tue 2021-05-04 05:31:06 -03; 2s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 7646 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCE
 Main PID: 7650 (apache2)
    Tasks: 6 (limit: 2267)
   Memory: 14.9M
   CGroup: /system.slice/apache2.service
           └─7650 /usr/sbin/apache2 -k start
             └─7651 /usr/sbin/apache2 -k start
               └─7652 /usr/sbin/apache2 -k start
                 └─7653 /usr/sbin/apache2 -k start
                   └─7654 /usr/sbin/apache2 -k start
                     └─7655 /usr/sbin/apache2 -k start

mai 04 05:31:06 WEBSERVER systemd[1]: Starting The Apache HTTP Server...
mai 04 05:31:06 WEBSERVER apachectl[7646]: AH00558: apache2: Could not reliably
mai 04 05:31:06 WEBSERVER systemd[1]: Started The Apache HTTP Server.
lines 1-19/19 (END)

```

Fonte: Autoria Própria

4.3 TESTES ZABBIX E GRAFANA

O Grafana possibilita a visualização dos dados obtidos com a coleta feita pelo Zabbix, desde que um *dashboard* seja devidamente configurado com as informações que deseja monitorar dos servidores. A Figura 30, apresenta o *dashboard* configurado, com os seguintes dados exibidos:

- Notificações do Zabbix;
- Espaço de armazenamento utilizado nos discos dos servidores;
- Uso de CPU e memória RAM nos servidores;
- Tráfego de rede no roteador.

Analisando a Figura 30 é possível verificar no painel superior os incidentes informados através do monitoramento do Zabbix. Ainda é possível verificar o monitoramento em tempo real dos discos dos servidores Webserver, Zabbix e AD. No monitoramento dos discos é possível notar em vermelho as partições “:” dos discos dos servidores WebServer e Zabbix, representando capacidade de armazenamento próxima do limite. Ainda Figura 30, é possível verificar a utilização de CPU e memória RAM nos servidores monitorados, em tempo real. Por último na parte inferior da Figura

30 é apresentado o tráfego de rede no Roteador que faz a comunicação de rede entre os servidores e a internet.

Figura 30 – Dashboard Configurado Grafana



Fonte: Autoria Própria

O *dashboard* do Grafana apresentou maior facilidade de configuração e adaptação dos itens monitorados em relação ao *dashboard* disponibilizado pelo Zabbix. Conforme é apresentado no capítulo 2.4.1, ele possui uma interface que possibilita melhor leitura dos recursos apresentados e maior personalização dos gráficos.

4.4 RESULTADOS OBTIDOS COM OS TESTES

Os testes realizados nas Seções 4.1 e 4.2 deste trabalho, mostram a capacidade do Zabbix em monitorar de forma eficiente os agentes de rede.

Com os eventos de alertas configurados no Zabbix e enviados ao Telegram, conforme foi citado na Seção 4.1, é possível verificar que os envios de alertas são feitos de forma rápida, no mesmo minuto que ocorreu o incidente, da mesma forma são informadas as resoluções de incidentes e atualizações no sistema.

Analisando os dados apresentados na seção 4.2 é possível notar que o Grafana integrado ao Zabbix possui capacidade de informar ao administrador de rede,

de forma visual, os recursos dos servidores monitorados em tempo real. Ainda são apresentados alertas que possibilitam a análise do ambiente a fim de proporcionar ações preditivas e corretivas no ambiente monitorado.

Comparando os resultados obtidos com o trabalho de BAHLS (2016) é possível verificar que, ambos possuem finalidades diferentes, porém se assemelham no intuito de fornecer soluções rápidas ao administrador da rede para que em caso de falhas o sistema possa ser normalizado com rapidez e eficiência.

No que se refere aos testes realizados e apresentados na Seção 4.2, esses apontam a necessidade de análise dos dados por parte do administrador de rede. Ao comparar os resultados obtidos na seção 4.2 com os resultados obtidos por BAHLS (2016), verifica-se que o ambiente testado por Bahls não possui monitoramento de dados em tempo real de forma visual (*dashboards*) o que pode dificultar o administrador da rede nas ações preditivas e na tomada de decisão.

Analisando os resultados obtidos por FONSECA (2018), é possível verificar que ambos possuem resultados similares, com alertas enviados através da aplicação Telegram, possibilitando a ação rápida nos incidentes por parte do administrador de rede.

5 CONCLUSÃO

Observando os aspectos analisados, concluem-se que o Zabbix é uma ferramenta de com grande acervo de documentação técnica, possibilitando a análise das ferramentas fornecidas pelo Zabbix.

De acordo com os testes realizados, foi possível verificar a eficácia do Zabbix no monitoramento reativo do ambiente de rede, com possibilidade de integração com diversas ferramentas. A integração do Zabbix ao Telegram possibilita de forma ágil, no mesmo minuto, o diagnóstico de possíveis problemas na rede, otimizando o tempo de resposta a incidentes por parte dos administradores de rede.

O Grafana possibilitou a visualização dos recursos do ambiente de rede de forma visual, auxiliando o administrador da rede com a análise dos recursos disponíveis, possibilitando ações preditivas a fim de evitar futuras indisponibilidades no ambiente monitorado.

Em virtude dos fatos mencionados conclui-se que o Zabbix atrelado ao Telegram e ao Grafana possibilita agilidade no monitoramento dos ativos, fornecendo soluções rápidas para solução de incidentes.

5.1 TRABALHOS FUTUROS

Para um melhor aproveitamento do potencial de monitoramento que ficou evidenciado após a comprovação dos resultados, o próximo passo para implementação do monitoramento poderá ser feito em uma rede de grande porte, possibilitando a análise dos dados obtidos com as ferramentas disponibilizadas, proporcionando o monitoramento proativo do ambiente de rede.

REFERÊNCIAS

BAHLS, A. **Monitoramento proativo do ambiente de rede utilizando o software Zabbix**. 2016. 64 f. Trabalho de Conclusão de Curso (Graduação em Análise e Desenvolvimento de Sistemas) - Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2016.

BONOMO, E. **Gerenciamento e monitoração de redes de Computadores utilizando-se Zabbix**. 2006. 68 f. Monografia (Pós-Graduação em Administração de Redes Linux) - Universidade Federal de Lavras. Minas Gerais, 2006.

CETIC.BR. Pesquisa sobre o uso das tecnologias de informação e comunicação nas empresas brasileiras. **TIC Empresas 2019**. 2019. Disponível em: <https://cetic.br/pt/tics/empresas/2019/empresas/A1B/>. Acesso em: 01 abr. 2021.

CISCO. **Cisco annual internet report (2018 – 2023)**. mar. 2020. Disponível em: <https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html>. Acesso em: 06 mai. 2021.

GRAFANA. **Documentação Grafana: Iniciando**. 2021. Disponível em: <https://grafana.com/docs/grafana/latest/getting-started>. Acesso em: 06 mai. 2021.

FONSECA, E. R. J. **Monitoramento de ambiente de rede utilizando Zabbix**. 6 maio. 2018. Disponível em: http://zabbixbrasil.org/files/Monitoramento_Ambiente_Red_Utilizando_Zabbix-Edilmar_Junior.pdf. Acesso em: 01 abr. 2021.

REMONTTI, R. Instalação do Zabbix 5 + notificações pelo Telegram nativo + Grafana 7 + Debian 10 Buster. **Blog Remontti**. 26 nov. 2020. Disponível em: <https://blog.remontti.com.br/4348>. Acesso em: 02 mai. 2021.

SOARES, A. S.; MATTOS, G. M.; ANTUNES, I. S.; COTA, R. E. **SIMPLE NETWORK MANAGEMENT PROTOCOL**. 2013. Disponível em: https://www.gta.ufrj.br/grad/10_1/snmp/contato.htm. Acesso em: 06 mai. 2021.

TELEGRAM. **Telegram FAQ**. 2021. Disponível em: <https://telegram.org/faq>. Acesso em: 05 mai. 2021.

ZABBIX. **Zabbix Documentation 5.2. Manual Zabbix**. 2021. Disponível em: <https://www.zabbix.com/documentation/current/pt/manual>. Acesso em: 02 mai. 2021.

LOPES, Raquel V.; SAUVÉ, Jacques P.; NICOLLETTI, Pedro S., **Melhores Práticas para Gerência de Redes de Computadores**. 1 ed. Rio de Janeiro: Campus, 2003.

APÊNDICE A – Instalação e Configuração da ferramenta Zabbix e da aplicação Grafana

Instalação e Configuração do Zabbix

1 – Aquisição do Zabbix

Os pacotes do Zabbix foram adquiridos através do site oficial do Zabbix pelo link “<https://www.zabbix.com/download>” e os arquivos no formato *.deb*, utilizados para o SO Debian. Após a instalação do repositório do Zabbix, a instalação dos pacotes é feita através do gerenciador de pacotes do Debian, o *Advanced Packaging Tool (APT)* (ZABBIX, 2021).

3 – Instalação dos pacotes Zabbix

Para instalação do Zabbix a partir dos pacotes, primeiro foi necessário a instalação do repositório do Zabbix, para possibilitar ao Debian localizar os pacotes para instalação. Para instalação do repositório, foi necessário a criação de um usuário com privilégios de root, para obter em seguida o repositório com o comando, “`# wget https://repo.zabbix.com/zabbix/5.2/debian/pool/main/z/zabbix-release/zabbix-release_5.2-1+debian10_all.deb`” (ZABBIX, 2021):

Após a execução do comando, o SO se conecta com o repositório do Zabbix e faz o download do repositório, conforme ilustra a Figura 31.

Figura 31 - Obtendo Repositório Zabbix

```

root@Zabbix:/home/zabbix# wget https://repo.zabbix.com/zabbix/5.0/debian/pool/main/z/zabbix-release/zabbix-releas
e_5.0-1+buster_all.deb
--2021-04-28 20:03:04-- https://repo.zabbix.com/zabbix/5.0/debian/pool/main/z/zabbix-release/zabbix-release_5.0-
1+buster_all.deb
Resolvendo repo.zabbix.com (repo.zabbix.com)... 178.128.6.101, 2604:a880:2:d0::2062:d001
Conectando-se a repo.zabbix.com (repo.zabbix.com)[178.128.6.101]:443... conectado.
A requisição HTTP foi enviada, aguardando resposta... 200 OK
Tamanho: 4236 (4,1K) [application/octet-stream]
Salvando em: "zabbix-release_5.0-1+buster_all.deb"

zabbix-release_5.0-1+buster_ 100%[=====] 4,14K --KB/s em 0s
2021-04-28 20:03:05 (59,2 MB/s) - "zabbix-release_5.0-1+buster_all.deb" salvo [4236/4236]

```

Fonte: Autoria Própria

O comando “`dpkg -i zabbix-release_5.2-1+debian10_all.deb`”, foi utilizado para instalar o repositório conforme ilustrado pela Figura 32 (ZABBIX, 2021).

Figura 32 - Instalando repositório do Zabbix

```

root@Zabbix:/home/zabbix# dpkg -i zabbix-release_5.0-1+buster_all.deb
A seleccionar pacote anteriormente não seleccionado zabbix-release.
(Lendo banco de dados ... 99345 ficheiros e directórios actualmente instalados.)
A preparar para desempacotar zabbix-release_5.0-1+buster_all.deb ...
A descompactar zabbix-release (1:5.0-1+buster) ...
Configurando zabbix-release (1:5.0-1+buster) ...

```

Fonte: Autoria Própria

O comando “*apt update*” atualiza a lista de repositórios para instalação, tornando o repositório Zabbix pronto para ser utilizado, conforme é ilustrado pela Figura 32 (ZABBIX, 2021).

Figura 33 - Atualizando Lista de Repositórios Zabbix

```

root@Zabbix:~# apt update
Atingido:1 http://deb.debian.org/debian buster InRelease
Obter:2 http://security.debian.org/debian-security buster/updates InRelease [65,4 kB]
Obter:3 http://deb.debian.org/debian buster-updates InRelease [51,9 kB]
Obter:4 http://security.debian.org/debian-security buster/updates/main Sources [182 kB]
Obter:5 http://security.debian.org/debian-security buster/updates/main amd64 Packages [284 kB]
Baixados 584 kB em 1s (755 kB/s)
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
All packages are up to date.
root@Zabbix:~# apt upgrade
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
Calculando atualização... Pronto
0 pacotes atualizados, 0 pacotes novos instalados, 0 a serem removidos e 0 não atualizados.

```

Fonte: Autoria Própria

O Comando “*apt install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-agent*” instala o Servidor Zabbix, a Interface web e o Agente Zabbix, conforme mostra a Figura 34 (ZABBIX, 2021).

Figura 34 - Instalação dos Pacotes Zabbix

```

root@Zabbix:/home/zabbix# apt install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-agent
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils fping libapache2-mod-php libapache2-mod-php7.3 libapr1
  libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libconfig-inifiles-perl libcurl4 libdbd-mysql-perl
  libdbi-perl liblua5.2-0 libodbc1 libopenipmi0 libssh-4 libterm-readkey-perl mariadb-client-10.3
  mariadb-client-core-10.3 php-bcmath php-common php-gd php-ldap php-mbstring php-mysql php-xml php7.3-bcmath
  php7.3-cli php7.3-common php7.3-gd php7.3-json php7.3-ldap php7.3-mbstring php7.3-mysql php7.3-opcache
  php7.3-readline php7.3-xml psmisc snmpd ssl-cert
Pacotes sugeridos:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom php-pear libclone-perl libmldbm-perl
  libnet-daemon-perl libsql-statement-perl libmyodbc odbc-postgresql tdsodbc unixodbc-bin snmptrapd
  openssl-blacklist zabbix-nginx-conf virtual-mysql-server
Os NOVOS pacotes a seguir serão instalados:
  apache2 apache2-bin apache2-data apache2-utils fping libapache2-mod-php libapache2-mod-php7.3 libapr1
  libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libconfig-inifiles-perl libcurl4 libdbd-mysql-perl
  libdbi-perl liblua5.2-0 libodbc1 libopenipmi0 libssh-4 libterm-readkey-perl mariadb-client-10.3
  mariadb-client-core-10.3 php-bcmath php-common php-gd php-ldap php-mbstring php-mysql php-xml php7.3-bcmath
  php7.3-cli php7.3-common php7.3-gd php7.3-json php7.3-ldap php7.3-mbstring php7.3-mysql php7.3-opcache
  php7.3-readline php7.3-xml psmisc snmpd ssl-cert zabbix-agent zabbix-apache-conf zabbix-frontend-php
  zabbix-server-mysql
0 pacotes atualizados, 47 pacotes novos instalados, 0 a serem removidos e 0 não atualizados.
É preciso baixar 23,1 MB de arquivos.
Depois desta operação, 114 MB adicionais de espaço em disco serão usados.
Você quer continuar? [S/n] s

```

Fonte: Autoria Própria

O Debian busca todos os pacotes necessários para a instalação dos serviços, solicitando a confirmação do usuário, para este passo foi necessário pressionar a tecla “s” e aguardar o término da instalação.

3 – Criação do banco de dados

Para o banco de dados foi utilizado o MariaDB, disponível no repositório oficial do Debian, a instalação do servidor MariaDB é feita pelo comando “*apt install mariadb-server-10.3*”, conforme apresenta 35.

Figura 35 - Instalação do servidor de Banco de dados MariaDB

```

root@Zabbix:/home/zabbix# apt install mariadb-server-10.3
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
 galera-3 gawk libcgi-fast-perl libcgi-pm-perl libfcgi-perl libhtml-template-perl libsigsegv2
 mariadb-server-core-10.3 rsync socat
Pacotes sugeridos:
 gawk-doc libipc-sharedcache-perl mailx mariadb-test netcat-openbsd tinyca openssh-server
Os NOVOS pacotes a seguir serão instalados:
 galera-3 gawk libcgi-fast-perl libcgi-pm-perl libfcgi-perl libhtml-template-perl libsigsegv2
 mariadb-server-10.3 mariadb-server-core-10.3 rsync socat
0 pacotes atualizados, 11 pacotes novos instalados, 0 a serem removidos e 0 não atualizados.
É preciso baixar 12,9 MB de arquivos.
Depois desta operação, 113 MB adicionais de espaço em disco serão usados.
Você quer continuar? [S/n] s

```

Fonte: Autoria Própria

Ao final os pacotes necessários para a instalação foram apresentados necessitando da confirmação do usuário, pressionando a tecla “s”.

Durante a instalação do Zabbix foi necessário criar a base de dados para a utilização da aplicação, ou seja, o banco de dados necessitava estar funcionando. Conforme o manual Zabbix (2021), a seguinte sequência de comando foi executada no terminal do SO para a criação do banco de dados:

```

#mysql -uroot -p
senha
mysql> create database zabbix character set utf8 collate utf8_bin;
mysql> create user zabbix@localhost identified by 'senha';
mysql> grant all privileges on zabbix.* to zabbix@localhost;
mysql> quit;

```

Na sequência, com o banco já criado, foi realizada a alimentação da base para o funcionamento da aplicação e novamente foi necessário acessar o terminal de comando do SO com um usuário *root* e executar o comando, “# *zcat /usr/share/doc/zabbix-server-mysql*/create.sql.gz | mysql -uzabbix -p zabbix*”. Este comando realiza a alimentação do banco de dados. O comando solicita a senha do usuário Zabbix, cadastrada no passo da criação do banco de dados e após este último passo a base de dados ficou pronta para ser utilizada pela aplicação (ZABBIX, 2021).

4 – Ajuste de Configuração do Servidor

Para que houvesse o funcionamento da aplicação, foram aplicados ajustes no arquivo de configuração do Zabbix. Abrindo o terminal de comando do SO com um usuário *root* e abrindo o arquivo “*zabbix_server.conf*” por meio comando, “*Nano /etc/zabbix/zabbix_server.conf*”. E novamente o comando exigiu a senha do banco de dados no campo “*DBpassword*”, conforme apresenta a Figura 36 (ZABBIX, 2021).

Figura 36 - Ajuste Arquivo de Configuração Zabbix

```
DBUser=zabbix
### Option: DBPassword
# Database password.
# Comment this line if no password is used.
#
# Mandatory: no
# Default:
DBPassword=zabbix█
```

Fonte: Autoria Própria

Também foram realizados, ajustes no Apache, informando o fuso horário da região em que o servidor será utilizado no arquivo de configuração, o comando, “*Nano /etc/zabbix/apache.conf*”, abre o arquivo de configuração. No arquivo, no campo, “*date.timezone*”, foi informado o fuso horário local, conforme ilustrado pela Figura 37 (ZABBIX, 2021).

Figura 37 - Ajuste Arquivo de Configuração Apache

```

<Directory "/usr/share/zabbix">
    Options FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all

    <IfModule mod_php5.c>
        php_value max_execution_time 300
        php_value memory_limit 128M
        php_value post_max_size 16M
        php_value upload_max_filesize 2M
        php_value max_input_time 300
        php_value max_input_vars 10000
        php_value always_populate_raw_post_data -1
        php_value date.timezone America/Sao_Paulo
    </IfModule>
    <IfModule mod_php7.c>
        php_value max_execution_time 300
        php_value memory_limit 128M
        php_value post_max_size 16M
        php_value upload_max_filesize 2M
        php_value max_input_time 300
        php_value max_input_vars 10000
        php_value always_populate_raw_post_data -1
        php_value date.timezone America/Sao_Paulo
    </IfModule>
</Directory>

```

Fonte: Autoria Própria

Após a realização das etapas acima e com todos os ajustes devidamente configurados, o servidor ficou pronto para ser iniciado. A Figura 38 mostra que foi necessário que o servidor Zabbix, o agente Zabbix e o Apache, fossem reiniciados e habilitados (ZABBIX, 2021).

Figura 38 - Iniciando Zabbix Server, Agente Zabbix e Apache

```

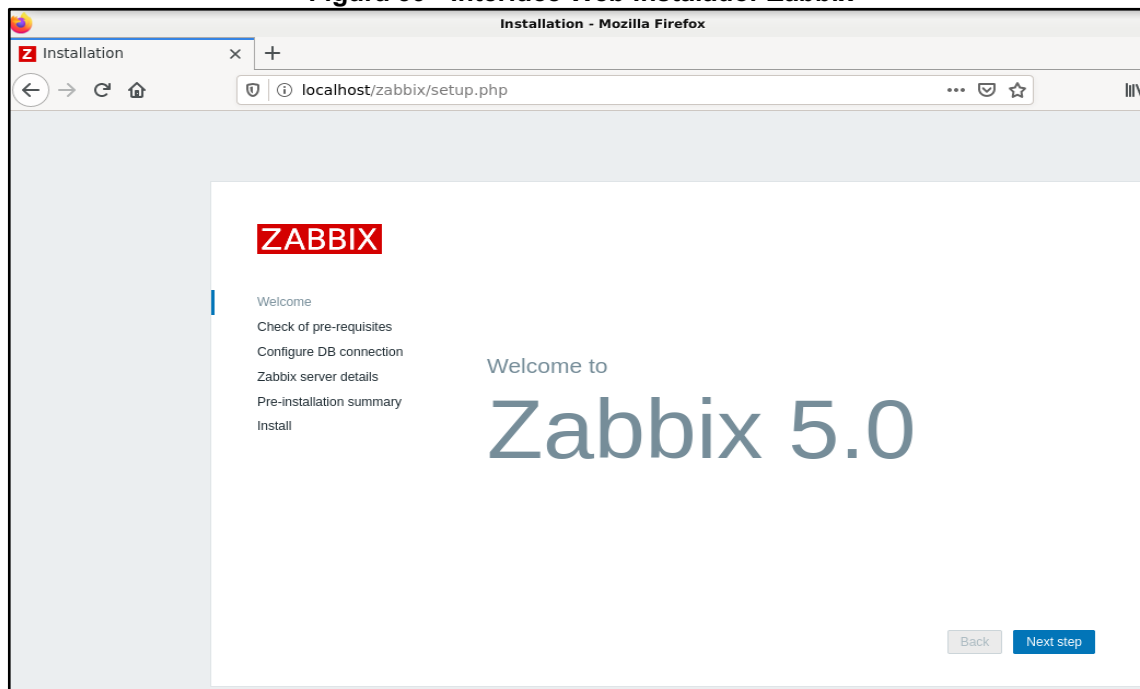
root@Zabbix:/home/zabbix# systemctl restart zabbix-server zabbix-agent apache2
root@Zabbix:/home/zabbix# systemctl enable zabbix-server zabbix-agent apache2
Synchronizing state of zabbix-server.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable zabbix-server
Synchronizing state of zabbix-agent.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable zabbix-agent
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
Created symlink /etc/systemd/system/multi-user.target.wants/zabbix-server.service → /lib/systemd/system/zabbix-se
rver.service.

```

Fonte: Autoria Própria

5 – Configuração Aplicação Zabbix

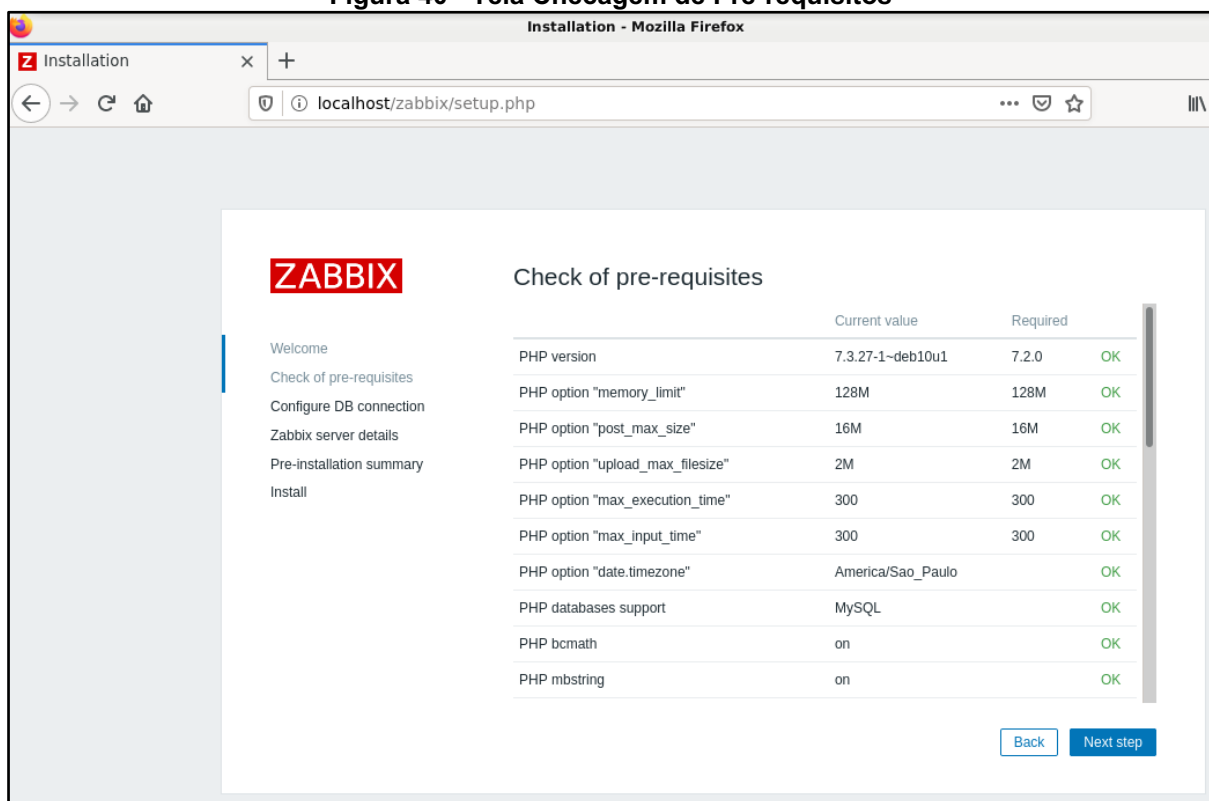
A configuração da aplicação do Zabbix foi realizada por meio da interface, disponibilizada no link "http://servidor_ip_ou_nome/zabbix" e acessada pelo navegador conforme apresenta a Figura 39.

Figura 39 - Interface Web Instalador Zabbix

Fonte: Autoria Própria

Primeiramente, clicou-se em “*Next step*”, onde foram exibidos os pré-requisitos do servidor. Nesta etapa foram checados, conforme Figura 40, se todos pré-requisitos estavam em correto funcionamento, caso algum problema fosse detectado seria necessário a resolução do mesmo para dar prosseguimento para as próximas etapas de configuração do Zabbix.

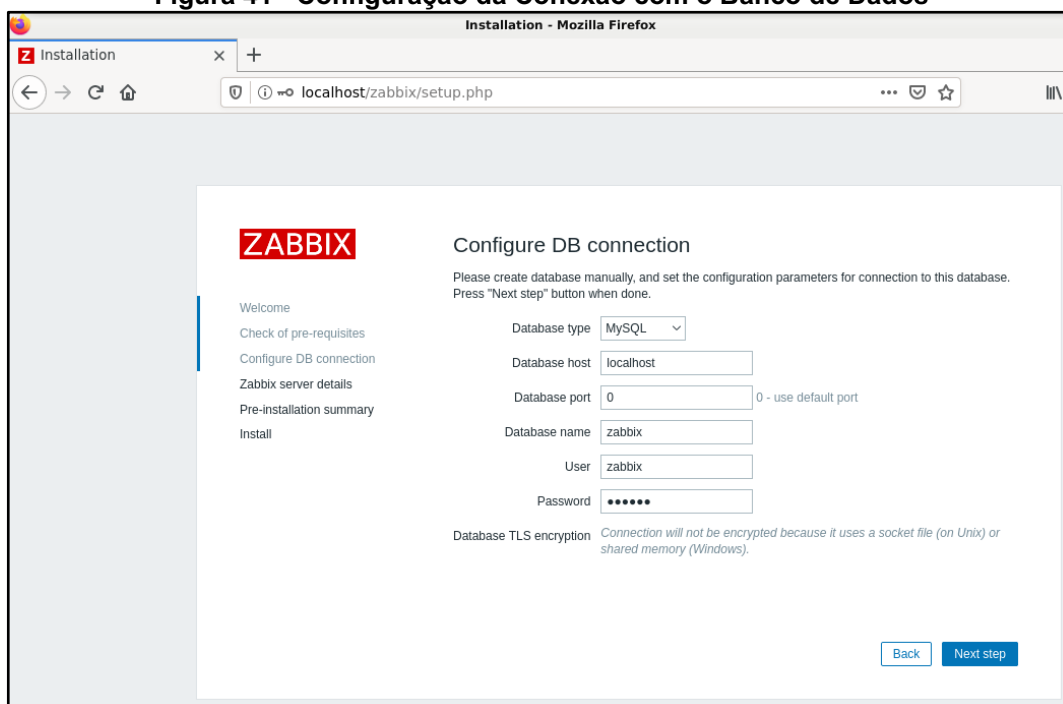
Figura 40 - Tela Checagem de Pré-requisitos



Fonte: Autoria Própria

Novamente, clicando em "Next step", para dar prosseguimento a configuração, foi apresentado pela interface do instalador a configuração da conexão do banco de dados. Na interface foram informados o tipo de banco de dados, o IP e a porta de funcionamento do servidor de banco de dados, o nome da base criada, o usuário e a senha de acesso ao banco de dados. Somente com o banco de dados em funcionamento, conforme apresenta Figura 41, foi possível configurar a conexão entre aplicação e o banco de dados.

Figura 41 - Configuração da Conexão com o Banco de Dados

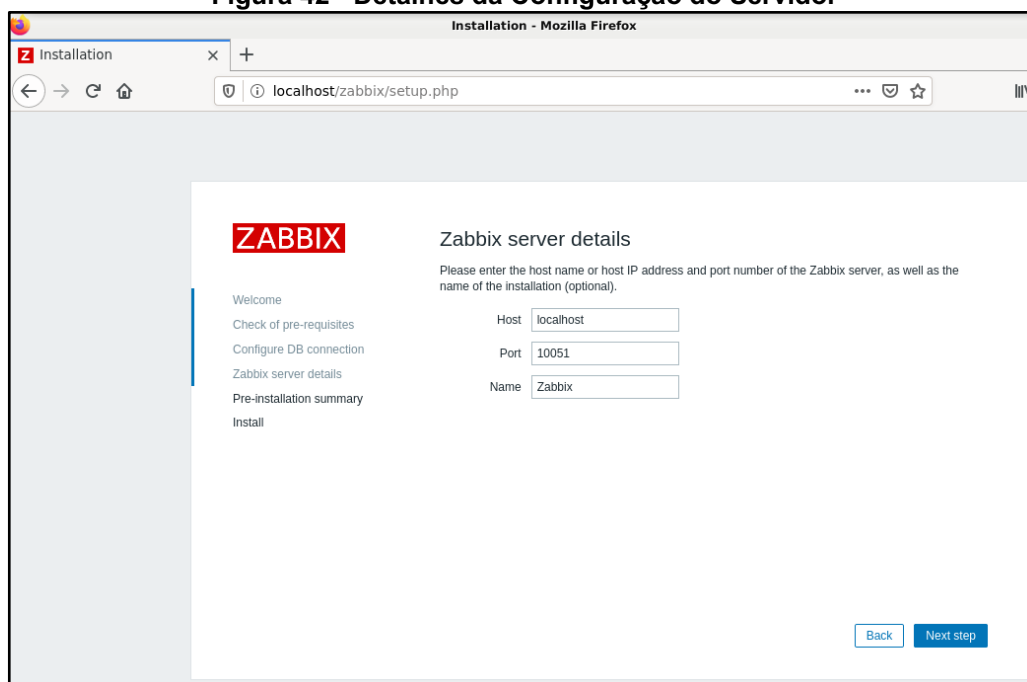


The screenshot shows the Zabbix installation web interface in a Mozilla Firefox browser. The browser's address bar displays 'localhost/zabbix/setup.php'. The page title is 'Installation - Mozilla Firefox'. The main content area is titled 'ZABBIX Configure DB connection'. On the left, a navigation menu lists the installation steps: Welcome, Check of pre-requisites, Configure DB connection (highlighted), Zabbix server details, Pre-installation summary, and Install. The main form contains the following fields: 'Database type' (a dropdown menu set to 'MySQL'), 'Database host' (text input with 'localhost'), 'Database port' (text input with '0', with a note '0 - use default port'), 'Database name' (text input with 'zabbix'), 'User' (text input with 'zabbix'), and 'Password' (password input field with masked characters). Below these fields, there is a note about 'Database TLS encryption' stating 'Connection will not be encrypted because it uses a socket file (on Unix) or shared memory (Windows)'. At the bottom right, there are 'Back' and 'Next step' buttons.

Fonte: Autoria Própria

Dando sequência, por meio do “Next step”, a interface de instalação exibe o nome do host, a porta (padrão 10051) e o nome de configuração, enfim todos os detalhes de configuração do Zabbix, conforme apresenta 42.

Figura 42 - Detalhes da Configuração do Servidor

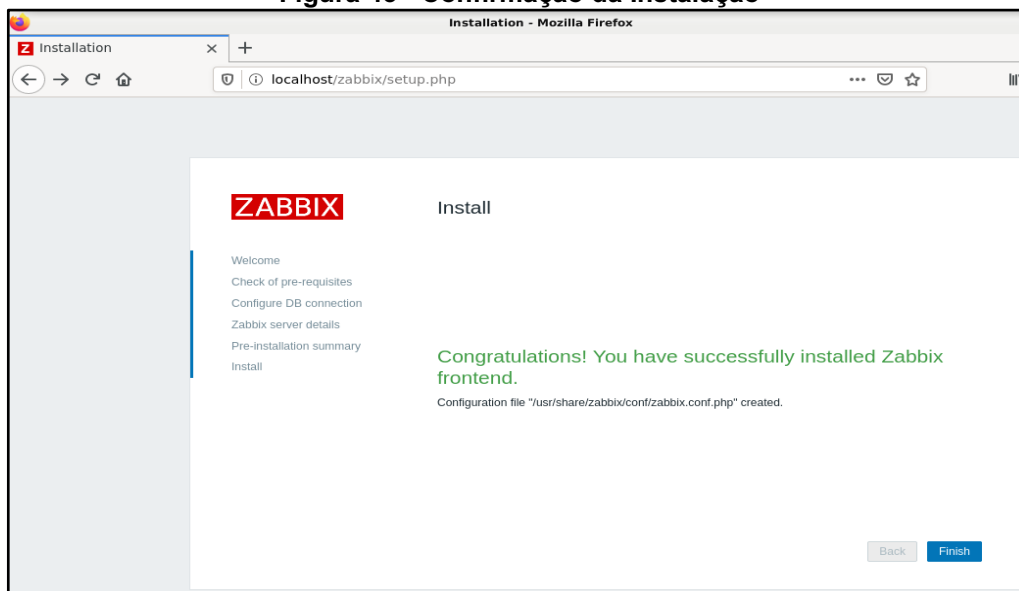


The screenshot shows the Zabbix installation web interface in a Mozilla Firefox browser. The browser's address bar displays 'localhost/zabbix/setup.php'. The page title is 'Installation - Mozilla Firefox'. The main content area is titled 'ZABBIX Zabbix server details'. On the left, a navigation menu lists the installation steps: Welcome, Check of pre-requisites, Configure DB connection, Zabbix server details (highlighted), Pre-installation summary, and Install. The main form contains the following fields: 'Host' (text input with 'localhost'), 'Port' (text input with '10051'), and 'Name' (text input with 'Zabbix'). Below these fields, there is a note: 'Please enter the host name or host IP address and port number of the Zabbix server, as well as the name of the installation (optional)'. At the bottom right, there are 'Back' and 'Next step' buttons.

Fonte: Autoria Própria

Por fim, a interface exibe se o processo de configuração do servidor Zabbix foi devidamente finalizado, a Figura 43 ilustra essa última mensagem da configuração da interface.

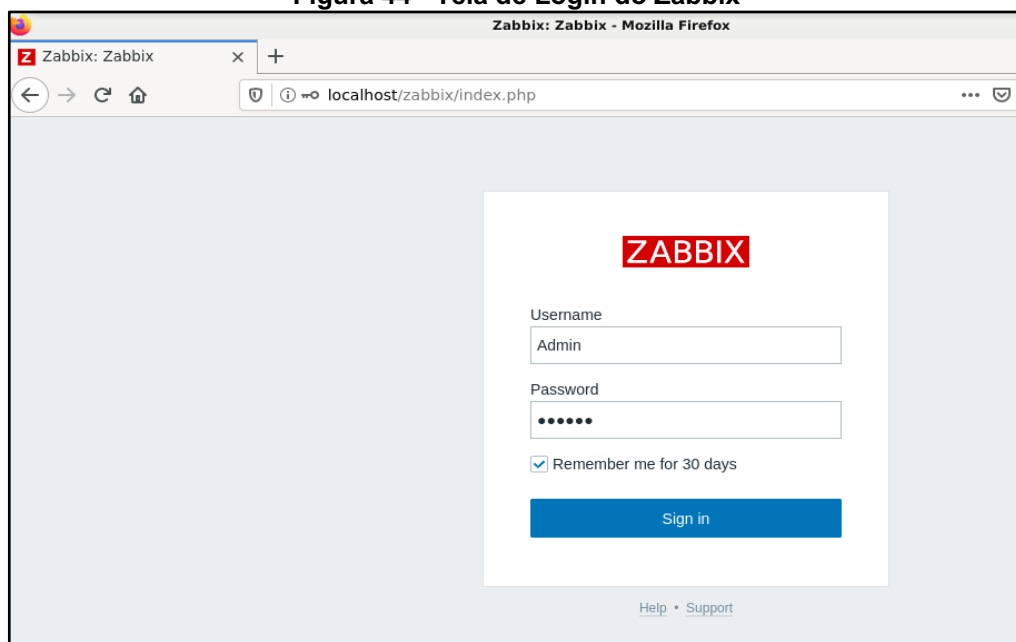
Figura 43 - Confirmação da Instalação



Fonte: Autoria Própria

Ao clicar em “*Finish*” a interface abre a tela de login do Zabbix, conforme exibido na Figura 44. No primeiro acesso o login padrão é o “Admin” e senha “zabbix”.

Figura 44 - Tela de Login do Zabbix



Fonte: Autoria Própria

No primeiro acesso ao servidor Zabbix observou-se que, o host do servidor já estava executando e realizando o monitoramento de suas atividades, essa configuração foi realizada de forma automática durante o processo de instalação do servidor, no momento em que foi adicionado o Zabbix Agente.

Instalação e Configuração do Grafana

1 – Aquisição do Grafana

A instalação do Grafana ocorreu primeiramente com a instalação dos pacotes que foram utilizados pela aplicação, a instalação foi realizada com comando “*apt install gnupg2 apt-transport-https software-properties-common*” (REMONTTI, 2020).

A Figura 45 ilustra a execução do comando para instalação dos pacotes, nesse passo foi necessário a confirmação do usuário, pressionando a tecla “s”.

Figura 45 - Instalação dos Pacotes Grafana

```

root@Zabbix:/home/zabbix# apt install gnupg2 apt-transport-https software-properties-common
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
  python3-distro-info python3-software-properties unattended-upgrades
Pacotes sugeridos:
  bsd-mailx default-mta | mail-transport-agent needrestart
Os NOVOS pacotes a seguir serão instalados:
  apt-transport-https gnupg2 python3-distro-info python3-software-properties software-properties-common
  unattended-upgrades
0 pacotes atualizados, 6 pacotes novos instalados, 0 a serem removidos e 0 não atualizados.
É preciso baixar 762 kB de arquivos.
Depois desta operação, 1.497 kB adicionais de espaço em disco serão usados.
Você quer continuar? [S/n] s

```

Fonte: Autoria Própria

Ainda, na instalação do Grafana foi preciso baixar a chave, Key, do repositório pelo comando “*# wget -q -O - https://packages.grafana.com/gpg.key | apt-key add*”, já para a adição do o repositório do Grafana foi inserido o comando, “*echo "deb https://packages.grafana.com/oss/deb stable main" | tee -a /etc/apt/sources.list.d/grafana.list*” (REMONTTI, 2020). A Figura 46, apresenta a execução dos comandos para adicionar o repositório do Grafana.

Figura 46 - Adicionando Repositório Grafana

```

root@Zabbix:/home/zabbix# wget -q -O - https://packages.grafana.com/gpg.key | apt-key add -
OK
root@Zabbix:/home/zabbix# echo "deb https://packages.grafana.com/oss/deb stable main" | tee -a /etc/apt/sources.l
ist.d/grafana.list
deb https://packages.grafana.com/oss/deb stable main

```

Fonte: Autoria Própria

2 – Instalação do Grafana e Plugin

Antes que se iniciasse de fato a instalação do Grafana foi necessário atualizar a lista dos repositórios, com o comando, “*apt update*”. Com o repositório atualizado foi iniciada a instalação do Grafana utilizando o comando, “*apt install grafana*” (REMONTTI, 2020). Na Figura 47 é possível verificar a execução do comando de atualização do repositório e instalação do Grafana, com a exigência da confirmação do usuário, pressionando a tecla “s”.

Figura 47 - Atualizando Repositórios e Instalando o Grafana

```

root@Zabbix:/home/zabbix# apt update
Atingido:1 http://deb.debian.org/debian buster InRelease
Obter:2 http://deb.debian.org/debian buster-updates InRelease [51,9 kB]
Atingido:3 http://security.debian.org/debian-security buster/updates InRelease
Atingido:4 http://repo.zabbix.com/zabbix/5.0/debian buster InRelease
Obter:5 https://packages.grafana.com/oss/deb stable InRelease [12,1 kB]
Obter:6 https://packages.grafana.com/oss/deb stable/main amd64 Packages [21,9 kB]
Baixados 86,0 kB em 1s (101 kB/s)
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
All packages are up to date.
root@Zabbix:/home/zabbix# apt install grafana
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
Os NOVOS pacotes a seguir serão instalados:
  grafana
0 pacotes atualizados, 1 pacotes novos instalados, 0 a serem removidos e 0 não atualizados.
É preciso baixar 52,3 MB de arquivos.
Depois desta operação, 178 MB adicionais de espaço em disco serão usados.

```

Fonte: Autoria Própria

A integração do Zabbix com Grafana foi realizada por meio do *plugin grafana-cli* instalado com o comando, “*grafana-cli plugins install alexanderzobnin-zabbix-app*” (REMONTTI, 2020). Na Figura 48, é possível observar a instalação do *plugin* de integração do Zabbix com Grafana.

Figura 48 – Instalação do Plugin de Integração do Grafana com Zabbix

```

root@Zabbix:/home/zabbix# grafana-cli plugins install alexanderzobnin-zabbix-app
installing alexanderzobnin-zabbix-app @ 4.1.4
from: https://grafana.com/api/plugins/alexanderzobnin-zabbix-app/versions/4.1.4/download
into: /var/lib/grafana/plugins
✓ Installed alexanderzobnin-zabbix-app successfully
Restart grafana after installing plugins . <service grafana-server restart>

```

Fonte: Autoria Própria

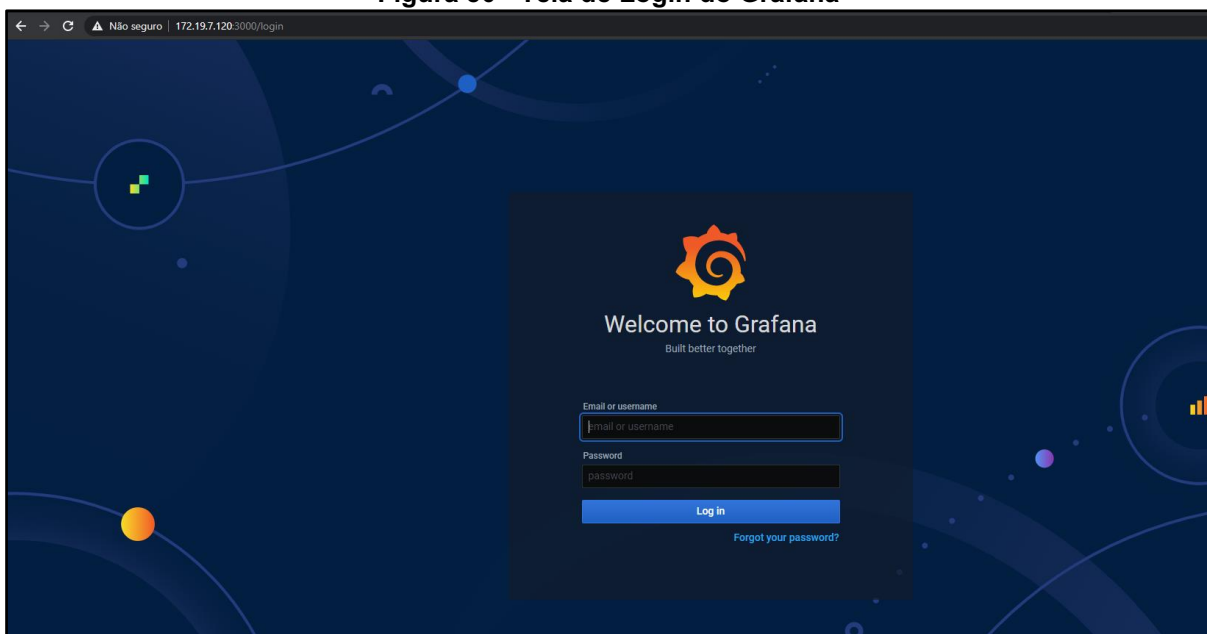
Para concluir a instalação e garantir início automático na inicialização do SO, foi necessário ativar e iniciar o servidor do Grafana e reiniciar o serviço *Daemon*, com a sequência de comandos “*# systemctl daemon-reload*”, “*# systemctl enable grafana-server*” e “*# systemctl start grafana-server*” (REMONTTI, 2020). A Figura 49, ilustra o passo a passo da execução dos comandos para início automático, ativação e inicialização do Grafana.

Figura 49 - Adicionando Grafana Início Automático, Ativando e Inicializando.

```
root@Zabbix:/home/zabbix# systemctl daemon-reload
root@Zabbix:/home/zabbix# systemctl enable grafana-server
Synchronizing state of grafana-server.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable grafana-server
Created symlink /etc/systemd/system/multi-user.target.wants/grafana-server.service → /lib/systemd/system/grafana-server.service.
root@Zabbix:/home/zabbix# systemctl start grafana-server
```

Fonte: Autoria Própria

Após o início do Grafana, foi possível acessá-lo pelo navegador, com o *link* “http://ip_servidor:3000”. O Grafana utiliza a porta “3000” como padrão e para o primeiro acesso, o usuário e a senha padrão é “*admin*”, sendo que foi necessária a troca da senha após o primeiro acesso (REMONTTI, 2020). A tela de login é exibida na Figura 50.

Figura 50 - Tela de Login do Grafana

Fonte: Autoria Própria