



UNIVERSIDADE FEDERAL DE GOIÁS

ESCOLA DE ENGENHARIA ELÉTRICA, MECÂNICA E DE COMPUTAÇÃO

**INTEGRAÇÃO DE UM SISTEMA DE MONITORAMENTO AOS MEDIDORES E
INVERSORES DE ENERGIA DA EMC**

RHUAN WEBSTER DE LOURENÇO E SILVA

RACHIMI SOLAREVISKY DE JESUS

ORIENTADOR: MARCELO STEHLING DE CASTRO

ARTIGO CIENTÍFICO DE PROJETO FINAL DE CURSO DE GRADUAÇÃO EM
ENGENHARIA DE COMPUTAÇÃO

GOIÂNIA, GO, BRASIL: 2024



UNIVERSIDADE FEDERAL DE GOIÁS
ESCOLA DE ENGENHARIA ELÉTRICA, MECÂNICA E DE COMPUTAÇÃO

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO PARA DISPONIBILIZAR VERSÕES ELETRÔNICAS DE TRABALHO DE CONCLUSÃO DE CURSO DE GRADUAÇÃO NO REPOSITÓRIO INSTITUCIONAL DA UFG

Na qualidade de titular dos direitos de autor, autorizo a Universidade Federal de Goiás (UFG) a disponibilizar, gratuitamente, por meio do Repositório Institucional (RI/UFG), regulamentado pela Resolução CEPEC no 1240/2014, sem ressarcimento dos direitos autorais, de acordo com a Lei no 9.610/98, o documento conforme permissões assinaladas abaixo, para fins de leitura, impressão e/ou download, a título de divulgação da produção científica brasileira, a partir desta data.

O conteúdo dos Trabalhos de Conclusão dos Cursos de Graduação disponibilizado no RI/UFG é de responsabilidade exclusiva dos autores. Ao encaminhar(em) o produto final, o(s) autor(a)(es)(as) e o(a) orientador(a) firmam o compromisso de que o trabalho não contém nenhuma violação de quaisquer direitos autorais ou outro direito de terceiros.

1. Identificação do Trabalho de Conclusão de Curso de Graduação (TCCG)

Nome(s) completo(s) do(a)(s) autor(a)(es)(as): Rhuan Webster de Lourenço e Silva e Rachimi Solarevisky de Jesus

Título do trabalho: Integração de um Sistema de Monitoramento aos medidores e inversores de energia da EMC

2. Informações de acesso ao documento (este campo deve ser preenchido pelo orientador) Concorda com a liberação total do documento [X] SIM [] NÃO¹

[1] Neste caso o documento será embargado por até um ano a partir da data de defesa. Após esse período, a possível disponibilização ocorrerá apenas mediante: a) consulta ao(à)(s) autor(a)(es)(as) e ao(à) orientador(a); b) novo Termo de Ciência e de Autorização (TECA) assinado e inserido no arquivo do TCCG. O documento não será disponibilizado durante o período de embargo.

Casos de embargo:

- Solicitação de registro de patente;
- Submissão de artigo em revista científica;
- Publicação como capítulo de livro.

Obs.: Este termo deve ser assinado no SEI pelo orientador e pelo autor.



Documento assinado eletronicamente por **Rhuan Webster De Lourenço E Silva , Discente**, em 02/08/2024, às 11:13, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Rachimí Solarevisky De Jesus, Discente**, em 02/08/2024, às 13:41, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Marcelo Stehling De Castro, Professor do Magistério Superior**, em 02/08/2024, às 15:46, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **4708463** e o código CRC **C27731BA**.

RHUAN WEBSTER DE LOURENÇO E SILVA

RACHIMI SOLAREVISKY DE JESUS

Integração de um Sistema de Monitoramento aos medidores e inversores de energia da EMC

Trabalho de conclusão de curso apresentado na Escola de Engenharia Elétrica, Mecânica e de Computação como requisito para a conclusão do curso de Engenharia de Computação e obtenção do título de Engenheiro de Computação.

Universidade Federal de Goiás - UFG

Escola de Engenharia Elétrica, Mecânica e de Computação - EMC

Orientador: Prof. Dr. Marcelo Stehling de Castro

Goiânia, Goiás, Brasil: 2024

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UFG.

Silva, Rhuan Webster de Lourenco e
Integração de um Sistema de Monitoramento aos medidores e inversores de Energia da EMC [manuscrito] / Rhuan Webster de Lourenco e Silva, Rachimi Solarevisky de Jesus. - 2024.
XVII, 17 f.: il.

Orientador: Prof. Dr. Marcelo Stehling de Castro.

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de Goiás, Escola de Engenharia Elétrica, Mecânica e de Computação (EMC), Engenharia da Computação, Goiânia, 2024.
Bibliografia.

Inclui abreviaturas, gráfico, tabelas.

1. Monitoramento. 2. Rede Eletrica. 3. Zabbix. 4. Grafana. 5. Telegram. I. Jesus, Rachimi Solarevisky de. II. Castro, Marcelo Stehling de, orient. III. Integração de um Sistema de Monitoramento aos medidores e inversores de Energia da EMC

CDU 62+004+005



UNIVERSIDADE FEDERAL DE GOIÁS
ESCOLA DE ENGENHARIA ELÉTRICA, MECÂNICA E DE COMPUTAÇÃO

ATA DE DEFESA DE TRABALHO DE CONCLUSÃO DE CURSO

Ao(s) **31** dia(s) do mês de **julho** do ano de **2024** iniciou-se a sessão pública de defesa do Trabalho de Conclusão de Curso (TCC) intitulado “Integração de um Sistema de Monitoramento aos medidores e inversores de energia da EMC”, de autoria de **Rhuan Webster de Lourenço e Silva** e **Rachimi Solarevisky de Jesus**, do curso de **Engenharia de Computação**, da **Escola de Engenharia Elétrica, Mecânica e de Computação (EMC)** da UFG. Os trabalhos foram instalados pelo Prof. Dr. **Marcelo Stehling de Castro** – EMC/UFG com a participação dos demais membros da Banca Examinadora: Prof. Dr. **Sérgio Granato de Araújo** – EMC/UFG e Engenheiro Especialista **Gustavo Dias de Oliveira** – EMC/UFG. Após a apresentação, a banca examinadora realizou a arguição do estudante. Posteriormente, de forma reservada, a Banca Examinadora atribuiu a nota final de **(9,0)**, tendo sido o TCC considerado **APROVADO**.

Proclamados os resultados, os trabalhos foram encerrados e, para constar, lavrou-se a presente ata que segue assinada pelos Membros da Banca Examinadora.



Documento assinado eletronicamente por **Marcelo Stehling De Castro, Professor do Magistério Superior**, em 02/08/2024, às 10:58, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Sergio Granato De Araujo, Professor do Magistério Superior**, em 02/08/2024, às 11:06, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Gustavo Dias De Oliveira, Técnico de Tecnologia da Informação**, em 02/08/2024, às 13:50, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **4674280** e o código CRC **851EB71D**.

Integração de um Sistema de Monitoramento aos medidores e inversores de Energia da EMC

Rhuan Webster de Lourenço e Silva¹, Rachimi Solarevisky de Jesus², Marcelo Stehling de Castro³

Universidade Federal de Goiás (UFG) - Escola de Engenharia Elétrica, Mecânica e de Computação (EMC) - Goiânia, Goiás, Brasil 74601-010, e-mails: rhuanwebster@discente.ufg.br¹, rachimi@discente.ufg.br², graduandos em Engenharia de Computação. mcastro@ufg.br³, Professor Associado - EMC/UFG.

Resumo — A transição energética tem apresentado um crescimento exponencial tanto no Brasil quanto no mundo, sendo uma das principais razões para esse aumento a economia de energia elétrica. Em paralelo a esse cenário, ano após ano, as indústrias de dispositivos IoT voltados para a energia elétrica têm reportado balancetes com aumentos significativos nas vendas. Após a instalação dos dispositivos, o monitoramento de todo o sistema torna-se crucial, pois é por meio dele que se verifica a estabilidade do sistema, a identificação de equipamentos defeituosos ou com dificuldades operacionais, além de possibilitar a gestão e análise da energia gerada e consumida. O objetivo deste artigo é analisar as soluções disponíveis no mercado para o monitoramento de dispositivos IoT e implementar a solução mais eficiente para monitorar o status operacional dos medidores e inversores de energia da Universidade Federal de Goiás (UFG). Após uma análise comparativa das diversas ferramentas, concluiu-se que o Zabbix é a melhor solução, proporcionando um acompanhamento unificado em uma única plataforma. Essa escolha permite reduzir o tempo necessário para manutenção, substituição e diagnóstico de problemas, além de gerar economias significativas.

Palavras-chaves — Monitoramento, rede elétrica, Zabbix, Grafana, Telegram.

Abstract — *The energy transition has shown exponential growth both in Brazil and around the world, one of the main reasons for this increase being electricity savings. In parallel to this scenario, year after year, IoT device industries focused on electrical energy have reported balance sheets with significant increases in sales. After installing the devices, monitoring the entire system becomes crucial, as it is through this that the stability of the system is verified, the identification of defective equipment or those with operational difficulties, in addition to enabling the management and analysis of the energy generated. and consumed. The objective of this article is to analyze the solutions available on the market for monitoring IoT devices and implement the most efficient solution to monitor the operational status of energy meters and inverters at the Federal University of Goiás (UFG). After a comparative analysis of the different tools, it was concluded that Zabbix is the best solution, providing unified monitoring on a single platform. This choice allows you to reduce the time required for maintenance, replacement and problem diagnosis, in addition to generating significant savings.*

Keywords — monitoring, electrical network, Zabbix, Grafana, Telegram.

I. INTRODUÇÃO

A transição energética tem se tornado um tema de crescente relevância tanto no Brasil quanto no cenário global nos últimos anos. Estudos recentes indicam um aumento significativo no uso de energia solar. Desde a invenção das células fotovoltaicas por Russell Shoemaker Ohl no início da década de 1950, essa tecnologia tem avançado substancialmente, pois esse tipo de energia alternativa impulsiona a produção de energia limpa em grande escala e gera economia nas despesas com energia elétrica. [1].

O investimento para a construção de usinas de energia solar ainda é elevado. Atualmente, existem linhas de crédito específicos para sua construção e o governo brasileiro implementa programas para tentar apoiar a instalação dos equipamentos, no qual pode-se citar o projeto (PL 624/23) que visa trocar a Tarifa Social de Energia Elétrica pela instalação de pequenos geradores de energia solar para famílias carentes [2].

Após a instalação, surge o desafio de monitorar o funcionamento e a eficiência total do sistema. Os dispositivos de energia solar e elétrica são majoritariamente dispositivos IoT (Internet das Coisas), permitindo a integração na rede de internet. Contudo, muitos fabricantes dificultam o monitoramento ao bloquear o acesso a integração a outros sistemas, pois mesmo que os protocolos usados são do tipo aberto a comunicação se torna proprietária, obrigando o consumidor a adquirir sistemas específicos da fabricante para obter dados completos.

Atenta a essa demanda, desde 2018, a Universidade Federal de Goiás (UFG) tem investido em equipamentos para a construção de suas usinas de energia solar. Tanto que atualmente, possui 24 usinas fotovoltaicas, que são responsáveis por gerar aproximadamente 36% da eletricidade consumida, reduzindo significativamente os gastos [3].

Complementar a essa iniciativa, ainda em 2018 foi lançado o programa "UFG Sustentável". Este programa tem como objetivo implementar práticas sustentáveis e reduzir os gastos com energia elétrica [4].

A Universidade Federal de Goiás (UFG) desempenha um papel fundamental no desenvolvimento científico e tecnológico do país. Desde a implantação das usinas fotovoltaicas em sua estrutura própria, a universidade tem conduzido pesquisas e monitorado medidores e inversores de energia elétrica.

As redes de monitoramento existentes foram adquiridas em conjunto com os dispositivos, resultando em sistemas variados de acordo com a marca dos equipamentos. Pensando nisso este artigo tem como objetivo apresentar diferentes ferramentas de

monitoramento disponíveis no mercado e propor um único sistema capaz de monitorar os status de todos dos medidores e inversores de energia elétrica da UFG. Este sistema proporcionará controle em tempo real e mitigará anomalias, facilitando a manutenção dos equipamentos.

II. MONITORAMENTO

O monitoramento de redes refere-se à prática de observar e analisar o tráfego de dados e o desempenho dos dispositivos em uma rede de internet [5]. O objetivo dessa atividade é assegurar a disponibilidade, confiabilidade, segurança e eficiência da rede, por meio da identificação de problemas, otimização de recursos e implementação de medidas preventivas para evitar falhas ou interrupções indesejadas [6].

No contexto da rede elétrica, o monitoramento envolve a coleta e análise de dados em tempo real para avaliar o desempenho, a eficiência e a segurança do sistema. Isso inclui o acompanhamento dos fluxos de energia, a detecção de falhas e a identificação de problemas potenciais antes que se tornem crises.

O avanço tecnológico desempenha um papel fundamental no monitoramento de redes elétricas. Sensores inteligentes, dispositivos de medição avançados e sistemas de comunicação permitem a coleta de dados em tempo real e a transmissão eficiente dessas informações para os centros de controle. Além disso, técnicas de análise de dados, como aprendizado de máquina e inteligência artificial, são empregadas para extrair insights valiosos dos vastos conjuntos de dados coletados.

A função primordial do monitoramento é acompanhar um serviço ou aplicação, gerando dados para a avaliação de seu desempenho. Além de fornecer alertas, essa prática facilita a centralização da coleta de dados e a transformação desses dados em gráficos simples e objetivos, permitindo ao administrador da rede monitorar o desempenho e os resultados das aplicações.

Para realizar o monitoramento de forma eficiente, é necessário adquirir ferramentas específicas projetadas para essa finalidade. Essas ferramentas são desenvolvidas para auxiliar no acompanhamento e análise de grandes volumes de dados em larga escala.

Existem várias razões pelas quais a monitorização da rede elétrica é importante. Os cinco principais benefícios são os seguintes [7]:

- Eficiência energética;
- Manutenção preventiva;
- Segurança;
- Gestão e análise de dados
- Sustentabilidade.

A. Eficiência energética

O contínuo monitoramento da rede elétrica permite a identificação de padrões de consumo energético e a detecção de áreas de desperdício. Esta prática possibilita a implementação de medidas de eficiência energética, tais como a substituição de equipamentos ineficientes, a otimização dos horários de uso de energia e a identificação de perdas de energia. O resultado é uma redução no consumo total de eletricidade e uma economia significativa nos custos de energia elétrica.

B. Manutenção preventiva

O monitoramento da rede elétrica possibilita a identificação de falhas e problemas em tempo real, permitindo a implementação de manutenção preventiva. Essa abordagem ajuda a evitar paralisações inesperadas, bem como os custos e o tempo de inatividade associados à interrupção do fornecimento de energia. Além disso, a manutenção preventiva prolonga a vida útil dos equipamentos elétricos, reduzindo a necessidade de substituições frequentes.

C. Segurança

A monitorização da rede elétrica contribui para a segurança dos ocupantes do ambiente ao permitir a detecção e prevenção de sobrecargas elétricas, curtos-circuitos e outros problemas que possam representar riscos de incêndio ou danos às pessoas e à propriedade.

D. Gestão e análise de dados

O monitoramento da rede elétrica, possibilita a coleta de dados detalhados sobre o consumo de energia e o desempenho dos sistemas elétricos. Esses dados podem ser analisados para identificar tendências, detectar áreas de melhoria e tomar decisões informadas sobre a gestão de energia. A análise desses dados também pode revelar oportunidades para otimizar a operação do ambiente e reduzir os custos de energia.

E. Sustentabilidade

O monitoramento da rede elétrica contribui para a sustentabilidade ambiental ao permitir uma gestão mais eficiente da energia. Ao identificar áreas de desperdício e implementar medidas de eficiência energética, o ambiente reduz sua pegada de carbono, ajudando a combater as mudanças climáticas.

III. SOFTWARES EXISTENTES

Nesta parte do artigo, serão apresentados alguns dos *softwares* mais usados em Monitoramento de Redes de comunicação de dados, conforme são mostrados na Tabela I, e após serão realizadas análises individuais ressaltando os pontos mais importantes para que um administrador de rede consiga monitorar a rede proposta.

Será dado enfoque em pontos como: facilitar correções e substituição de equipamentos; Identificação de oportunidades de melhorias; Performance da rede; Análise e coleta de dados para documentação.

TABELA I
TABELA DE COMPARAÇÃO DOS SISTEMAS ESCOLHIDOS PARA O ESTUDO

Características	Ckeckmk	Zabbix	CloudStats
Open Source	Sim	Sim	Não
Console web	Sim	Sim	Sim
Administração web	Sim	Sim	Sim
Monitoramento de serviços	Sim	Sim	Sim
Relatórios de disponibilidade	Sim	Sim	Plugin
Coleta de dados SNMP	Sim	Sim	Sim
Monitoramento de Recursos	Sim	Sim	Sim
Mapas de rede	Sim	Sim	Plugin
Classificação de tráfego de Redes	Não	Plugin	Plugin
Coleta de Network Flows	Não	Não	Não
Deteção de violação de segurança	Não	Não	Não

Fonte: [6].

A. Checkmk

Checkmk é um *software* que foi desenvolvido em Python e C++ com o intuito de monitorar infraestrutura de TI [8]. Então podemos definir que checkmk é uma plataforma de monitoramento de TI que visa entregar uma rede escalável, automatizada e extensível pois através dele é possível obter uma visão completa de toda a infraestrutura de TI.

A ferramenta consegue oferecer monitoramento para aplicativos, servidores e redes locais ou na nuvem, além de contar com uma ampla gama de recursos, tornando assim possível monitorar sem esforço.

Atualmente a o software conta com um plano gratuito no qual está incluso dashboards, mas é uma versão muito simples e dependendo do segmento que será monitorado é necessário fazer upgrade, podendo custar até R\$4.000,00 por ano.

B. CloudStats

O CloudStats é uma plataforma de monitoramento de servidores e sites, através da ferramenta é possível monitor todos os recursos de qualquer servidor dedicado ou na nuvem que seja baseado em Linux e Windows.

Dentro as variadas funcionalidades se destacam a possibilidade de monitorar URLs e obter alerta como por exemplo a data de expiração do SSL ou de qualquer outro problema que possa estar ocorrendo [9].

Em julho de 2024, a ferramenta possui uma avaliação gratuita, mas após um período é cobrado US\$ 5 por mês para cada servidor que estiver sendo monitorado com todos os recursos disponíveis, e US\$ 3 por mês para o monitoramento de IP ou site.

C. Zabbix

O Zabbix é um sistema de monitoramento open source maduro e de classe empresarial, projetado para monitorar milhões de métricas. Entre as métricas monitoradas estão o desempenho do servidor, a disponibilidade do servidor e as alterações de configuração [10].

D. Spiceworks

O Spiceworks é uma plataforma abrangente de gerenciamento de TI, projetada para facilitar o monitoramento e a administração de redes e sistemas em ambientes empresariais [11]. A solução oferece uma variedade de recursos, incluindo inventário de ativos, monitoramento de rede, *help desk*, além de possuir uma comunidade online para compartilhamento de conhecimentos e suporte.

Uma das características da plataforma é simplificar tarefas como a detecção de dispositivos na rede, rastreamento de mudanças de configuração e gestão de tickets de suporte. Sua abordagem integrada visa proporcionar uma solução centralizada para as necessidades de TI, promovendo eficiência operacional e colaboração entre profissionais de TI.

E. Splunk

O Splunk é uma poderosa plataforma de análise de dados que se destaca no monitoramento e análise de grandes volumes de dados de diferentes fontes [12].

A solução permite que as organizações colem, indexem e analisem dados de máquinas, aplicativos e sistemas para obter insights valiosos.

Destaca-se sua capacidade de correlacionar dados de diferentes fontes, fornecendo uma visão holística do ambiente de TI e suas aplicações abrangem desde segurança da informação até monitoramento de desempenho, facilitando a identificação de padrões, tendências e potenciais problemas.

Além disso, o Splunk oferece recursos avançados de pesquisa, visualização e geração de relatórios para apoiar a tomada de decisões informadas nas organizações.

IV. REDE IOT DE MEDIDORES DE ENERGIA ELÉTRICA DA UFG

Atualmente a UFG possui vários modelos de dispositivos IOTs para medição e controle de energia elétrica, sendo eles de diferentes modelos e marcas.

A Quadro I com a descrição de modelo juntamente com sua característica técnica de comunicação via rede e por fim será apresentado um resumo de cada dispositivo constando sua importância e características.

Quadro I
TABELA DE COMPARAÇÃO DOS EQUIPAMENTOS IOTS

Equipamento	Características
CCK4400ME-TC	Protocolo Modbus/TCP IP
CCK 6700 E	Protocolo Modbus/TCP IP
CCK 7550 SF	Protocolo Modbus/TCP IP
HUAWEY SUN2000	Protocolo Modbus/TCP IP
WEG GTW001	Protocolo Modbus/TCP IP

Fonte: Autoria própria.

O CCK4400ME-TC é um transdutor de energia com memória de massa, Ethernet e TC Externo. É conhecido no mercado por ser compacto e oferecer uma variedade de funcionalidade avançada para medição e monitoramento de consumo de energia [13].



Fig. 1. CCK4400.

Fonte: [13].

O CCK6700E é um Gerenciador de energia multifuncional muito usado para medir, monitorar e gerenciar o consumo de energia de empresas ou indústrias [13]. Ao contrário do CCK4400ME, é conhecido no mercado por ser uma solução robusta.



Fig. 2. CCK6400E.

Fonte: [13].

O CCK7550S é um multimetro de grandezas elétricas e harmônicas com ethernet muito usado para medir, monitorar e gerenciar a qualidade da energia que está sendo recebida [13].



Fig. 3. CCK7550S.
Fonte: [13].

Os Huawei SUN2000 é uma linha de inversores solares que são fabricados pela Huawei Technologies. Esses inversores são projetados para transformar a energia elétrica gerada por painéis solares em energia que pode ser usada em residência ou alimentar a rede elétrica [14].

O Huawei SUN2000-33KTL-A se destaca no mercado por ser eficiente e bastante confiável, tanto que atualmente é usado em instalações residenciais, comerciais e industriais.



Fig. 4. Inversor Huawei SUN2000-33KTL-A.
Fonte: [14].

Os Gateway Weg na sua essência têm como característica a conversão de protocolos, com o decorrer do tempo esse tipo de dispositivo foi evoluindo, sendo capaz de realizar conexão com variados dispositivos que usam diferentes softwares sem contar que por conta do seu elevado desempenho é possível ser usado como microprocessador uma vez que através dele é possível realizar processamento de dados local [15].

O Gateway WEG GTW001-SIW realiza as leituras com os dispositivos/inversores de campo, sua forma de conectividade é através de uma rede RS-485, protocolo Modbus-RTU;

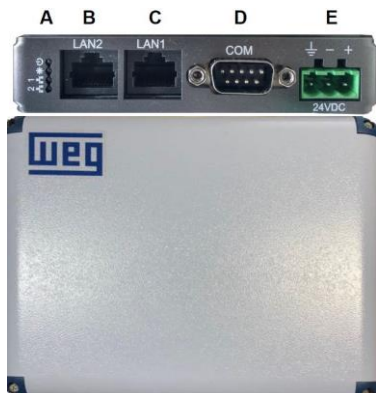


Fig. 5. Gateway WEG GTW001-SIW.
Fonte: [15].

V. INSTALAÇÃO E CONFIGURAÇÃO DOS SOFTWARES

Dos softwares analisados no capítulo III conclui-se que o Zabbix é o mais interessante para ser implementado por conta da solução ser open source, mais usada mundialmente, possuir uma comunidade bastante ativa que está em crescente evolução e proporciona ao administrador a liberdade de poder desenvolver suas ideias. Nesta seção será dado enfoque ao Zabbix e posteriormente será descrito toda a instalação.

Zabbix é um *software* de monitoramento de rede criado por Alexei Vladishev, visando atender ao público empresarial [10]. Desde sua criação até os dias atuais a ferramenta se destaca pela grande abrangência de parâmetros que é suportado monitorar, sem contar que é totalmente gratuita, uma vez que o Zabbix é escrito e distribuído sob a Licença Pública Geral GPL versão 2.

Essa solução é bastante poderosa e usual nos dias de hoje, pois além de poder proporcionar um monitoramento de rede altamente integrado, conta com vários recursos dentro de um único pacote, que serão listados abaixo:

- Coleta de dados altamente completa podendo suportar protocolos SNMP, IPMI, JMX dentre outros.
- Variadas integrações com softwares de mensagem onde se torna possível configurá-los facilmente para implementação do sistema de alerta de problemas.
- Possui funcionalidade gráfica integrada para exibição de gráficos em tempo real, mas também possui integração com variados softwares como o Grafana para uma implementação de diferentes tipos de gráficos, mapa de rede, relatórios dentre outros
- Altamente seguro pois possui autenticação de usuário, além de poder limitar determinadas visualizações
- Interface web bem intuitiva e rápida, possibilitando assim acesso facilitado de qualquer lugar.
- Descoberta de rede automatizada, proporcionando assim realizar uma descoberta de dispositivos de rede ou registro de agentes de forma automática, além de conseguir descobrir até sistemas de arquivos, interfaces de rede e OIDs SNMP.
- Compatibilidade com variados bancos de dados como MySQL, MariaDB dentre outros.
- Possui o proxy Zabbix que é usado para monitoramento remoto, ou seja, a ferramenta está pronta para ambientes complexos
- Compatibilidade com variados servidores como Linux, IBM AIX, dentre outros.
- Coleta de dados via servidor/proxy e por agentes onde o agente é altamente completo podendo ser implementado em Linux e Windows além de ser facilmente extensível

A. Requisitos de plataforma para uso do Zabbix

Nesta seção será feito a descrição de todos os requisitos que devem ser observados e executados para o Zabbix funcionar na sua excelência e são especificamente 3(três): Hardware; Software; e Interface Web.

1. Hardware

- Memória: Para instalação do Zabbix é necessário haver memória física e de disco, sendo que a quantidade mínima depende da quantidade de host que será monitorado [10]. Por padrão da plataforma os requisitos mínimos de memória é 128MB e para armazenamento 256MB em disco.
- CPU: O uso de recursos da CPU também depende da quantidade de host que será monitorando, mas nesse caso deve se levar em consideração os mecanismos de banco de dados, pois o Zabbix e de forma especial o banco de dados Zabbix, dependendo das aplicações irão consumir bastante CPU, o QUADRO II está mostrando o tipo de CPU relacionada com cada hardware que possa ser escolhido.

QUADRO II

QUADRO DE REQUISITOS MÍNIMOS PARA A INSTALAÇÃO DO ZABBIX

Tamanho da Instalação	Métricas monitoradas	Núcleos de CPU/vCPU	Memória (GB)	Base de Dados
Pequeno	1.000,00	2	8	Mysql, Percona, Maria DB ou PostgreSql
Médio	10.000,00	4	16	Mysql, Percona, Maria DB ou PostgreSql
Grande	100.000,00	16	64	Mysql, Percona, Maria DB ou PostgreSql
Muito Grande	1.000.000,00	32	96	Mysql, Percona, Maria DB ou PostgreSql

Fonte: [10].

2. Software

O Zabbix é construído em torno de 3(três) pilares: Servidores web modernos, melhores banco de dados do mercado e linguagem de script PHP [10], o QUADRO III está mostrando a listagem de Banco de Dados que são mais usuais atualmente, facilitando assim a escolha.

QUADRO III

QUADRO DOS TIPOS DE BANCO DE DADOS QUE SÃO SUPORTADO NO ZABBIX

Programas	Versão
MySQL/Percona	8.0.30-8.4.X
Maria DB	10.5.00-11.4.X
Oracle	19c – 21c
PostgreSQL	13.0 - 16.X
TimescaleDB para PostgreSQL	2.1.0 – 2.15.X

Fonte: Autoria própria.

3. Interface Web

A interface Web é construída através de um servidor *front-end* para hospedagem web [10], o quadro IV está mostrando a listagem com os disponíveis no mercado facilitando assim a escolha.

QUADRO IV

QUADRO PARA ESCOLHA DO SISTEMA FRONT-END DO ZABBIX

Programas	Status obrigatório	Versão
Apache	Pelo menos um desses	2.4 ou posterior
Nginx		1.2 ou posterior
PHP	Sim	7.4.0 -8.3.x

Fonte: [10].

B. Alguns recursos disponíveis no Zabbix

1. Configuração Centralizada

Para obter uma configuração centralizada é necessário que o servidor e os dispositivos que serão monitorados estejam na mesma rede e atualmente o Zabbix disponibiliza esse recurso, pois a plataforma conta com o componente central que é o *server* Zabbix.

Esse tipo de configuração não é indicado quando a quantidade de host é bastante extensa pois o *server* pode sofrer bastante processamento simultâneos e nesses casos a plataforma orienta a utilizar servidores auxiliares uma vez que assim é proporcionado o balanceamento de carga de trabalho.

2. Variados tipos de monitoramento

A plataforma disponibiliza a possibilidade de variados tipos de monitoramento onde os principais são:

- **Monitoramento de disponibilidade:** É usado quando se quer garantir que serviços, aplicativos e sistemas de TI estejam operacionais e acessíveis aos usuários conforme esperado. A monitoração contínua desses recursos é necessária para identificar e resolver problemas rapidamente, reduzindo o tempo de inatividade.
- **Monitoramento de integridade:** É escolhido quando se quer verificar e garantir que os dados, sistemas e serviços de TI estejam operando corretamente e que não tenham sido alterados ou corrompidos. Esse tipo de monitoramento se concentra em manter a precisão, consistência e confiabilidade dos dados e sistemas ao longo do tempo.

3. Segurança na coleta de dados e na Gerência

Toda a coleta de dados dos dispositivos e guardada em um banco de dados, atualmente a plataforma suporta diferentes tipos de banco como por exemplo Oracle, MySQL, PostgreSQL e MariaDB e a configuração para a criação desses bancos seguem um passo a passo detalhado e criterioso de instalação e configuração.

O Zabbix disponibiliza para o administrador a criação de diferentes usuários, onde é possível definir diferentes níveis de acesso e caso alguma tentativa de acesso não seja autorizada é recebido notificações [10].

4. Relatórios

A plataforma disponibiliza a possibilidade do administrador gerar diferentes relatórios onde os principais são:

- relatório de disponibilidade: retorna à disponibilidade de serviços e dispositivos monitorados;
- relatório de desempenho: mostra o desempenho de sistema e aplicativos;
- relatório de eventos: retorna todas as ocorrências de eventos como falhas, alertas e ações corretivas;
- relatório de Inventário: apresenta a quantidade de hardware e software dos dispositivos monitorados.

Além de permitir personalizar o seu próprio relatório.

C. Componentes do Zabbix

A seguir, serão apresentados os principais componentes do Zabbix.: Zabbix Server; Zabbix Proxy; Zabbix Agente; Zabbix Agente 2 e Interface Web.

1. Zabbix Server

O servidor Zabbix exerce uma função primordial e central no software Zabbix, desempenhando uma variedade de tarefas cruciais. Através dele é possível coletar dados, computar gatilhos e emitir alertas aos usuários [10].

Além disso, atua como o ponto principal para a recepção de dados dos agentes e proxy Zabbix, que comunicam informações sobre a disponibilidade e integridade dos sistemas monitorados.

2. Zabbix Proxy

Atuando essencialmente em nome do *server* Zabbix, o Proxy Zabbix pode coletar informações de monitoramento de um ou mais dispositivos e as envia para o servidor [10].

Por padrão da plataforma os dados coletados são armazenados de forma temporária local antes de serem transferidos para o *server* Zabbix, por isso é importante deixar registrado que cada proxy precisa de um banco de dados separado que é onde ficará armazenado esses dados locais para posteriormente enviá-los ao server.

Por fim sua utilidade é opcional no pacote da plataforma, mas pode ser extremamente benéfico caso a demanda exija distribuir uma possível sobrecarga de trabalho do *server*, uma vez que com o *proxy* coletando os dados, o server principal vai estar consumindo menos recurso da CPU e do disco para processar as informações.

3. Zabbix Agente

O agente Zabbix é uma funcionalidade que é instalado em um host que será monitorado com o intuito de verificar recursos e aplicações locais, como discos rígidos, memória e estatísticas do processador [10]. Sua forma de trabalho é similar ao Zabbix Proxy, mas ele reúne informações operacionais no próprio dispositivo e transmite esses dados ao servidor para processamento.

4. Zabbix Agente 2

O agente Zabbix 2 é uma atualização do agente Zabbix, podendo até substituir o agente original pois ele foi projetado para esse fim [10], sem contar que possui as funcionalidades que serão apresentadas a seguir:

- Minimizar o número de conexões TCP

- Refinar a simultaneidade das verificações.
- Ser prontamente ampliável com plug-in capazes de executar verificações simples ou complexas.

5. Interface Web

A interface web do Zabbix é o front-end da ferramenta, nela é possível o administrador ter acesso ao *server* Zabbix na web, o único requisito que é necessário para o uso dessa funcionalidade é ter acesso a rede internet [10].

Sendo assim resumidamente a interface web é a parte visual e funcional do Zabbix, nesse componente o usuário consegue acessar todo o servidor realizando ações como por exemplo alterar as configurações, receber informações dentre outras.

D. Configurações de implantação

Nesta seção será descrito todo o processo de instalação das ferramentas que serão necessárias para que o projeto de monitoramento dos medidores de energia da UFG seja possível.

Serão utilizadas as plataformas Zabbix, Grafana e o Telegram pois as mesmas trabalhando em conjunto acredita-se que irá fornecer maior escalabilidade, flexibilidade, alertas e gestão de toda a rede.

Pensando em facilitar a manutenção posterior a entrega desse projeto, o servidor Zabbix foi instalado no Proxmox da Gerência de Redes da EMC. Esse ambiente foi preparado com o Debian 12, que é um sistema operacional Linux conforme o Zabbix exige, lembrando que o SO precisa fornecer um usuário com permissão administrador para as instalações pertinentes serem realizadas e conexão com a internet.

1) Instalação do MariaDB no Ubuntu/Debian

A primeira fase de instalação do Zabbix e a escolha do banco de dados e sua devida implementação seguindo todos os passos recomendados.

Passo 01: Para iniciar a instalação do MariaDB no Ubuntu/Debian é preciso digitar no terminal os comandos [16]:

```
sudo apt update
```

```
sudo apt install mariadb-server mariadb-client
```

Passo 02: Feito a instalação parte-se para a configuração do banco e o procedimento a se fazer é executar o script de segurança que irá executar as configurações básicas recomendadas que é feito com o seguinte comando no terminal [16]:

```
sudo mysql_secure_installation
```

Passo 03: Autenticar no banco de dados com o usuário root e a senha do sistema operacional e esse procedimento é feito no terminal através do comando no terminal [16]:

```
sudo mysql -u root -p
```

Passo 04: É o momento da criação do usuário e da senha de administrador do MariaDB, e é feito através dos seguintes comandos no console do MySQL [16]:

Create DATABASE zabbix_db character set utf8 collate utf8_bin; Comando esse que está criando uma tabela no banco de dados chamado Zabbix_bd e configurado as horas para a do Brasil.

CREATE USER 'zabbix_user'@'localhost' IDENTIFIED BY 'sua_senha'; Comando esse que está criando o usuário administrador e a senha para acessar as tabelas que acabaram de ser criadas.

GRANT ALL PRIVILEGES ON zabbix_user'@'localhost'; Comando esse que passa privilégio total para o usuário que acabou de ser criado.

FLUSH PRIVILEGES; Comando esse que está renovando os privilégios do banco de dados

EXIT; E por fim esse comando fecha – se o console do MySQL.

Passo 05: Como boas práticas se tratando de configurações de banco de dados é preciso reiniciar o servidor do MariaDB para validação se tudo foi configurado corretamente e para ser executado é preciso digitar o seguinte comando no terminal do debian:

```
sudo systemctl restart mariadb
```

2) Instalação do servidor web Apache

A segunda ferramenta instalada é o servidor web apache, uma vez que o mesmo é responsável por estabelecer uma conexão entre o servidor e a página web do Zabbix [17].

Passo 01: O primeiro procedimento é a instalação do servidor web Apache e no terminal do sistema operacional é preciso inserir o seguinte comando:

```
sudo apt install apache2
```

Passo 02: Como validação do serviço que acabou de ser instalado é necessário reiniciá-lo e para isso ainda no terminal digita-se o comando:

```
systemctl restart apache2
```

Passo 03: Com a instalação do apache concluída é preciso instalar as bibliotecas do php para suportar o front-end do Zabbix e a instalação é feita no terminal com o seguinte comando:

```
sudo apt install php7.0 libapache2-mod-php7.0.
```

3) Instalação do Zabbix

Neste momento, com todas as dependências instaladas, inicia-se a instalação do Zabbix.

Passo 01: O passo inicial é a instalação das dependências do Zabbix e para executa é necessário digitar o seguinte comando no terminal do Debian [16]:

```
sudo wget
https://repo.zabbix.com/zabbix/6.4/debian/pool/main/z/zabbix-
release_6.4-1+debian12_all.deb
```

Passo 02: Verifique se o pacote .deb foi importando junto com as dependências do Zabbix, pois o mesmo deve estar dentro do diretório para a instalação ser concluída. O procedimento a se fazer ainda no terminal é rodar o comando `ls`.

Passo 03: Com a confirmação do pacote .deb é preciso rodar o seguinte comando no terminal do Debian para finalizar a instalação [16]:

```
sudo dpkg -i zabbixrelease_release_6.41+debian12_all.deb.
```

Passo 04: Com a instalação concluída de todos os pacotes parte-se para a instalação dos módulos de funcionamento do Zabbix que são o server, frontend e agente. Para executá-los digita-se o seguinte comando no terminal [16]:

```
sudo install install zabbix-server-mysql zabbix-frontend-php
zabbix-apache-conf zabbix-sql-scripts zabbix-agent.
```

Passo 05: Com os módulos instalados é preciso fazer a validação do banco de dados junto ao Zabbix logo digita-se o seguinte comando no terminal do sistema operacional:

```
zcat /usr/share/zabbix-sql-scripts/mysql/server.sql.gz | mysql -
-default-character-set=utf8mb4 -u zabbix -p zabbix
```

Obs: Será solicitado a senha que foi criada no passo 04 da instalação do MariaDb.

Passo 06: Validação dos recursos que acabaram de ser instalados e sendo assim é necessário reiniciar e habilitar a conexão. Para executar o procedimento, no terminal do Debian digita-se:

```
systemctl restart zabbix-server zabbix-agent apache2
```

```
systemctl enable zabbix-server zabbix-agent apache2.
```

Passo 07: Por fim com o ambiente finalmente instalado, é preciso digitar o seguinte comando:

```
sudo ifconfig
```

Este comando identificará o IP atribuído ao servidor Zabbix e com essa informação basta abrir qualquer navegador de internet e inserir a seguinte url: `ipdoservidor/Zabbix` para conseguir acessar a página inicial de configuração do front-end.

No primeiro acesso é aberto a tela conforme a Fig. 6 está ilustrando, e assim é iniciado as configurações iniciais do *front-end* como, por exemplo, se todos os pré-requisitos e a conexão com o banco de dados estão em operação, dentre outros.

Nessa fase o administrador deverá apenas ir preenchendo com as informações corretas e ir clicando em “Next step”, ao todo serão 4 (quatro) telas de configurações que serão discriminadas a seguir [16].



Fig. 6. Tela de configuração inicial do Zabbix.
Fonte: Autoria própria.

A primeira tela que irá ser apresentada é a página de verificação de pré requisitos, é preciso apenas verificar se está tudo marcado com a palavra OK na cor Verde e caso esteja clicar em “Next step”.

A segunda é a integração com o Banco de dados, nessa fase coloca-se a configuração do nome da *database*, usuário e a senha que foi definida no passo 4 da instalação do Maria DB, inserindo corretamente as informações é preciso clicar em “Next Step” para avançar.

A terceira é feita a definição do nome do servidor, o horário e o tema que são todos personalizáveis pelo administrador, inserindo as informações clica-se em “Next Step” novamente.

A quarta é mostrada todas configurações que foram inseridas, o administrador deverá revisar se tudo está correto e caso esteja clicar em “Next Step” para finalizar assim a configuração. Irá ser retornando uma mensagem que a configuração foi concluída.

Com todos os requisitos instalado a página final é o dashboard principal do sistema Zabbix conforme é ilustrado na Fig. 7.

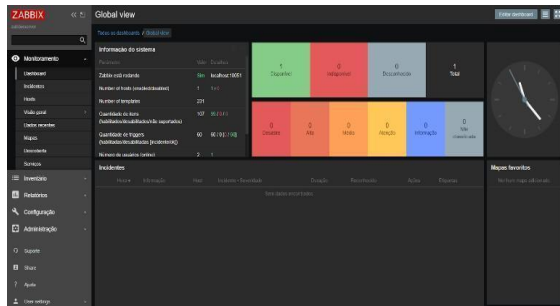


Fig. 7. Tela Inicial Zabbix.
Fonte: Autoria própria.

4) Criação do group

O group no Zabbix é definido como um conjunto de host e sua funcionalidade é bastante usada quando se precisa formar grupos de host para aplicar determinada ação ou quando se demanda atribuir direito de acesso a host para diferentes grupos de usuários [10].

A configuração é intuitiva, clica na opção Data collection > host Groups, irá abrir a tela com todos os groups criados até o momento e para adicionar mais um, o usuário deverá ir ao canto superior direito é clicar na opção create host group, e informa o único parâmetro solicitado que é o Group nome.

5) Criação do Host

O host é definido como um dispositivo, aplicação ou serviço que possa ser monitorado [10]. A sua fase de criação dentro do servidor é bem intuitiva, acessa o front end, clica na opção “monitoring” do menu e seleciona a opção “host”.

Irá abrir a tela com todos os hosts cadastrados até o momento e, para criar um novo host, usar no canto superior direito a opção “create host”, clica nela e irá abrir outra tela com todas as configurações possíveis, por padrão os seguintes campos são obrigatórios.

- Host Name: Para definir o nome do Host
- Host groups: Para definir qual grupo o determinado host pertence.

Embora não seja obrigatório, esses dois campos abaixo são importantes para o bom andamento do projeto.

- Template: Para aprimorar as coletas de dados do host e nesse projeto foi preciso adicionar o template “ICMP ping” para realizar as coletas sem interrupção de cada host da rede.
- Add: Para inserir o endereço ip do host.

Cada host cadastrado possui a aba inventário onde é possível completar informação dos dispositivos como marca, modelo, cidade e dentre essas opções. A latitude e longitude do dispositivo chama a atenção pois, através do cadastro dessas informações, pode-se adicionar o ponto no geomap que fica no dashboard de monitoramento, viabilizando assim o administrador em obter condições de verificar no mapa a localização real de cada host.

6) Criação de Item

O Item é definido como uma coleta de dados específica, que precisará ser coletado pelo Zabbix de um host, portanto sem item não há monitoramento [10].

Seu cadastro pode ser criado manual pelo usuário ou inserido automaticamente no host, nesse é preciso associar um template durante a configuração inicial do host.

Para a configuração individual o usuário deverá clicar na opção *monitoring* > Host, selecionar o host que será configurado e escolher a opção itens, em sequência irá abrir a tela com todos os itens cadastrados até o momento e para criar um novo deverá clica na opção *create item*.

Por padrão, os campos obrigatórios de preenchimento são Name, Key, Host interface, update interval history storage period e Trend storage period.

7) Criação do map

Uma prática essencial para melhorar a visualização e o entendimento do desempenho e da integridade da infraestrutura monitorada no Zabbix é a criação de mapas.

Através dele é possível ter uma visualização gráfica da topologia da rede, é possível identificar problemas facilmente, fazer uma análise de desempenho rigorosa e ágil, organizar e estruturar de forma lógica a rede e ter uma documentação viva uma vez que mapa é definido como uma forma dinâmica de documentação [10].

A sua criação no Zabbix acontece da seguinte forma: *monitoring* > *maps* > *create map*. Dessa forma irá abrir um diagrama onde o administrador poderá adicionar elementos, link, relacionamentos e movimentá-los da forma que desejar deixando a ferramenta com uma customização avançada.

8) Resultado da configuração

No escopo do projeto, a previsão era tornar possível monitorar todos os dispositivos IoTs, independente do modelo ou marca dos medidores e na Fig. 8 está destacado a criação de cada host existente.

Vale destaque para a coluna de *Problems* onde caso esteja escrito apenas a palavra *Problems* o host está sem nenhum tipo de problema e caso tenha a marcação de um numeral o medidor está informando que realmente aconteceu algum problema.

Name	Interface	Availability	Tags	Status	Latest data	Problems
Medidor_CCK4400_Agencia-de-Inovacao_Samambaia	192.168.0.60:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	Problems
Medidor_CCK4400_Agronomia_Samambaia	192.168.0.27:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	Problems
Medidor_CCK4400_Artes-Visuais_Samambaia	192.168.0.7:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	Problems
Medidor_CCK4400_BC-Inv-01_Samambaia	192.168.0.91:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_BC-Inv-01_Samambaia_test	192.168.0.91:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_BC-Inv-02_Samambaia	192.168.0.92:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_BC-Inv-03_Samambaia	192.168.0.93:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_BC-Inv-04_Samambaia	192.168.0.94:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	Problems
Medidor_CCK4400_BC-Inv-05_Samambaia	192.168.0.95:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_BC-Inv-06_Samambaia	192.168.0.96:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_Biblioteca-Central-Geral_Samambaia	192.168.0.8:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	Problems
Medidor_CCK4400_CAB_Samambaia	192.168.0.33:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	Problems
Medidor_CCK4400_CAC-Creche_Samambaia	192.168.0.31:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	Problems
Medidor_CCK4400_CE-CE QFAC 01-CEQFAC 02-Arvore-Solar_Samambaia	192.168.0.41:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_CE-Inv-01_Samambaia	192.168.0.81:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_CE-Inv-02_Samambaia	192.168.0.82:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_CE-Inv-03_Samambaia	192.168.0.83:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_CE-Inv-04_Samambaia	192.168.0.84:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_CE-Inv-05_Samambaia	192.168.0.85:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_CE-Inv-06_Samambaia	192.168.0.86:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_CE-Inv-07_Samambaia	192.168.0.87:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	1
Medidor_CCK4400_CE-Inv-08_Samambaia	192.168.0.88:10050	ZBX SNMP	class: network target: icmp	Enabled	Latest data 3	Problems

Fig. 8. Listagem de Host cadastrados.
Fonte: Autoria própria.

Para acessar as métricas que a plataforma Zabbix está recebendo o usuário deverá seguir o seguinte caminho, *monitoring > latest data*, selecione o *host* que deseja ver as informações e clique na opção *apply*, logo após será exibido uma página conforme a Fig. 9 está demonstrando, assim será possível verificar cada coleta de dados do host por item cadastrado, que no caso do projeto é *ICMP loss*, *ICMP ping* e *ICMP response time*.

Host	Name	Last check	Last value	Change	Tags
Gateway_WEG_14	ICMP: ICMP loss	14s	0 %		component: health component: network
Gateway_WEG_14	ICMP: ICMP ping	14s	Up (1)		component: health component: network
Gateway_WEG_14	ICMP: ICMP response time	14s	0.87ms	+0.098ms	component: health component: network

Fig. 9. Tela de *Latest Data*.
Fonte: Autoria própria.

Ainda na mesma página do *Latest Data*, na coluna *info* pode-se verificar que a plataforma Zabbix fornece um gráfico de cada métrica conforme as Fig. 10, 11 e 12 estão demonstrando. Nessa parte é exposto a forma gráfica da plataforma, mas acredita-se que falta a plataforma poder mostrar um gráfico unificado de todos os hosts deixando assim o recurso mais interessante.

Os gráficos são personalizáveis nos requisitos de tempo uma vez que o administrador pode filtrar o período que deverá ser apresentado e esse período pode ser configurado para até os últimos dois anos.

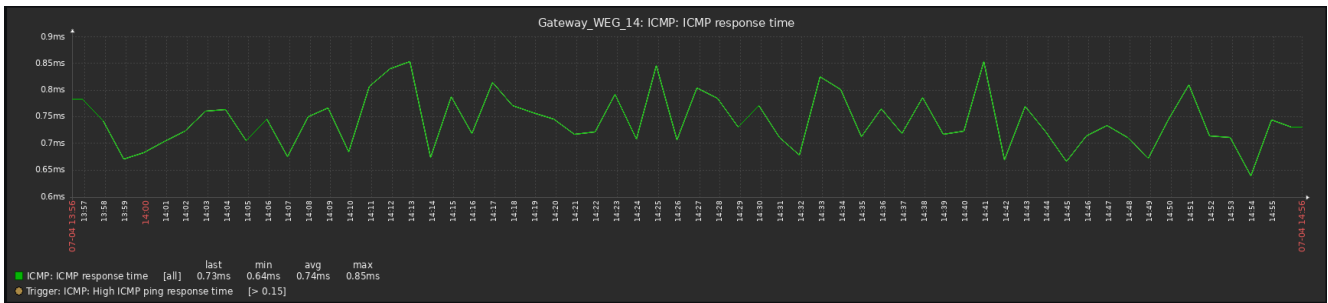


Fig. 10. Gráfico do Item ICMP response time.

Fonte: Autoria própria.

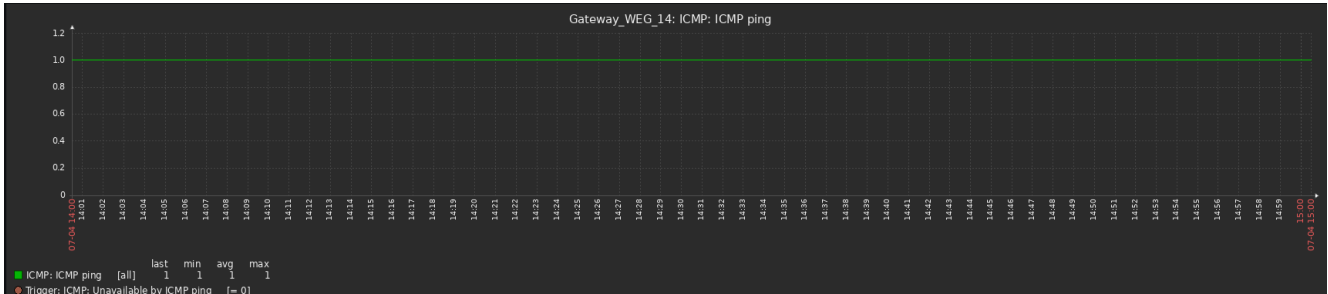


Fig. 11. Gráfico do Item ICMP ping.

Fonte: Autoria própria.

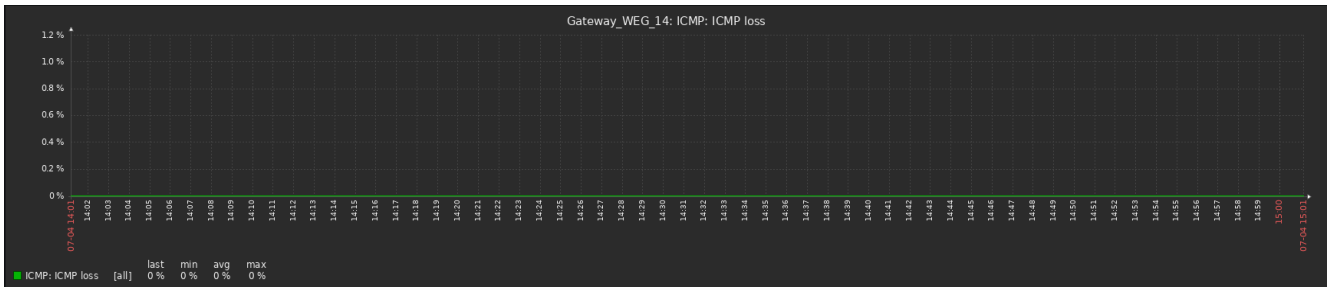


Fig. 12. Gráfico do Item ICMP loss.

Fonte: Autoria própria.

Como uma funcionalidade de suporte, o Zabbix oferece a capacidade de criar mapas que auxiliam o administrador na visualização e manutenção da rede monitorada. Esses mapas proporcionam uma representação gráfica da estrutura da rede, facilitando a compreensão e a gestão dos componentes monitorados, conforme ilustrado na Fig. 13.

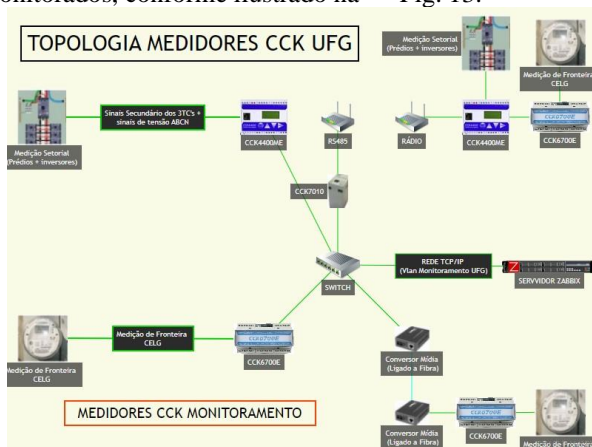


Fig. 13. Mapa com relacionamento entre o host e o servidor.

Fonte: Autoria própria.

E. Grafana

Grafana é uma plataforma de código aberto que permite a visualização, análise e monitoramento de dados [18]. Atualmente é uma solução popular e bastante utilizada pelos profissionais de TI, DevOps e desenvolvimento de software visando monitorar infraestrutura, aplicação e sistemas.

Dentre suas características algumas bastante importantes e serão discriminadas abaixo:

- **Usuários e permissões:** A plataforma conta com um eficiente sistema de gestão de usuários e permissões, disponibilizando assim a criação de vários login com diferentes níveis de acesso.
- **Comunidade e Suporte:** O Grafana é um projeto de código aberto e tem uma grande comunidade de usuários e desenvolvedores, bem como muitos fóruns de suporte e documentação.
- **Integração com diferentes fontes de Dados:** Atualmente a plataforma já suporta uma gama bem grande de fonte de dados e pode-se destacar Elasticsearch, MySQL, PostgreSQL entre outras

- Visualização de Dados: O Grafana permite a criação de painéis personalizados que mostram gráficos, tabelas e alertas baseados em dados de várias fontes.
- Plugins e Extensibilidade: O ecossistema de plugins da plataforma permite a inclusão de novas funcionalidades, integrações e painéis, além de permitir que os usuários possam desenvolver plugins.
- Alertas: O Grafana permite a configuração de alertas que são ativados quando os dados atendem a condições específicas. Esses alertas podem ser enviados por vários canais, incluindo Slack, PagerDuty e e-mail.
- Consultas Flexíveis: A plataforma possui um editor de consultas avançado permitindo a criação de consultas complexas e a manipulação flexível de dados.
- Uso empresarial: O Grafana possui a versão gratuita, mas para quem possui uma demanda empresarial a plataforma tem o Grafana Labs que disponibiliza alguns recursos adicionais como suporte aprimorado, segurança avançada e integração com outras ferramentas empresariais.

1) Instalação do Grafana no Debian 12

Passo 01: O primeiro passo é baixar a biblioteca e adicionar o usuário que dará suporte ao Grafana, logo após é necessário fazer o download no repositório do Grafana e buscar os pacotes necessários para a instalação.

Ao final deve-se fazer a instalação do pacote baixado no repositório. Todos esses passos são executados no terminal do Debian e para realizar lós e digitado os seguintes comandos:

```
sudo apt-get install -y adduser libfontconfig1 musl
```

```
wget https://dl.grafana.com/enterprise/release/grafana-enterprise_10.4.2_amd64.
```

```
sudo dpkg -i Grafana-enterprise_10.4.2_amd64.deb
```

Passo 02: Com a instalação concluída, deve se partir para iniciar o serviço, habilitar o server e por fim habilitar o front-end do serviço Grafana. E seguindo a mesma lógica do passo 01 todos esses passos são executados no terminal do Debian e para realizar lós e digitado os seguintes comandos:

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable grafana-server
```

```
systemctl start grafana-server
```

Passo 03: Com a fase de instalação e inicialização do Grafana finalizado, parte-se para a fase integração e nela é necessário instalar alguns plugins que são responsáveis pela integração com o Zabbix. Para realiza-lo digita-se o seguinte comando no terminal do Debian:

```
grafana-cli plugins install alexanderzobnin-zabbix-app
```

Passo 04: E por fim para finalizar a fase de instalação/integração, como medida de boas práticas em servidores, faz-se necessário reiniciar o serviço para que a solução reconheça o plugin recentemente instalado, uma vez que ele é o principal responsável pelo elo Grafana/Zabbix. Para realizar esse passo é preciso digitar o seguinte comando no terminal do Debian:

```
systemctl restart grafana-server.
```

Passo 05: Para acessar a Interface Web do Grafana recém instalada é preciso do ip do servidor e para identifica-lo usa-se o seguinte comando no terminal nativo do Debian:

```
sudo ifconfig
```

```
zhean@iot:~$ sudo ifconfig
ens18: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 200.137.220.102 netmask 255.255.255.0 broadcast 200.137.220.255
inet6 fe80::e8da:4bff:fe5b:9e47 prefixlen 64 scopeid 0x20<link>
ether sa:da:4b:4b:9e:e7 txqueuelen 1000 (Ethernet)
RX packets 164586571 bytes 11574739615 (10.7 GiB)
RX errors 0 dropped 222418 overruns 0 frame 0
TX packets 71792060 bytes 8276815296 (7.7 GiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Fig. 14. Tela de consulta do IP do servidor.

Fonte: Autoria própria.

Nesse caso, conforme ilustrado na Fig. 14, o ip do Grafana é o 200.137.220.102. Logo, para acessar a página de autenticação no navegador, é preciso digitar a seguinte url: [http://\(ipdografana\):3000](http://(ipdografana):3000), e o resultado que deverá ser alcançado é ilustrado na Fig. 15.

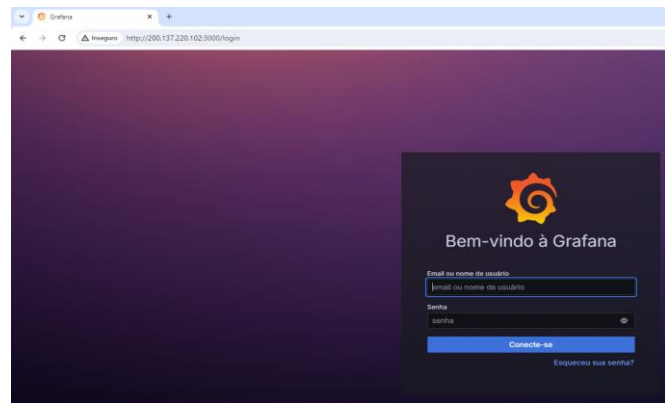


Fig. 15. Página inicial de autenticação do usuário no Grafana.

Fonte: Autoria própria.

Por padrão o primeiro acesso o usuário é admin e a senha admin e após a autenticação será solicitado o cadastro da senha definitiva que ficará a cargo do administrador definir.

2) Configuração para integração Grafana/Zabbix

A integração Grafana/Zabbix é totalmente configurada no Grafana. Na tela inicial da plataforma, nas opções da barra lateral e preciso selecionar a opção conexões, logo após irá aparecer duas opções: adicionar nova conexão e fonte de dados, deverá clicar na opção fonte de dados.

Aberto a tela com todas as fontes de dados adicionadas até o momento, no canto superior direito haverá um ícone em azul com os dizeres “Adicionar nova Fontes de dados”, é preciso clicar nela e pesquisar “Zabbix” que vai ser retornado por conta

da instalação do plugin realizado no Passo 03 da instalação.

A Fig. 16 mostra a tela de configuração da nova fonte de dados que será adicionada, que nesse caso é o Zabbix, nela é preciso configurar alguns parâmetros:

- **Name:** É um campo que fica à escolha do administrador para definir o nome da fonte de dados.
- **URL:** É preciso seguir o padrão pré-fixado, que sempre deverá ser informado https://localhost/api_jsonrpc.php, onde o localhost é o site de acesso ao front-end do Zabbix.
Exemplo: A interface web do Zabbix é acessada pelo iot.emc.ufg.br. Logo a URL que deverá ser digitada na página de configuração deverá ser https://iot.emc.ufg.br/api_jsonrpc.php.
- **Conexão Zabbix:** nesse campo é preciso informar as credenciais de acesso administrador do Zabbix, pois a API do Grafana vai comunicar com a API do Zabbix, trazendo assim todos os hosts cadastrados.

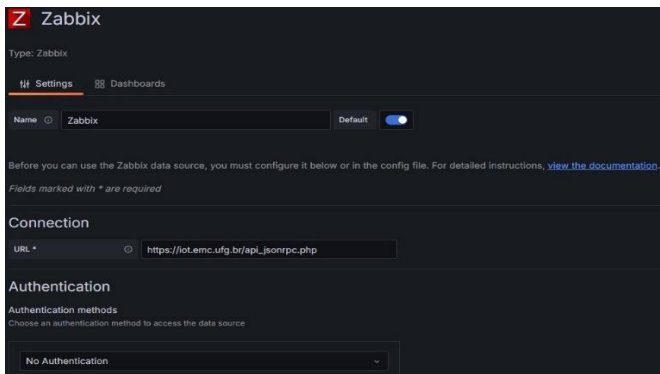


Fig. 16. Tela de configuração da integração do Grafana com o Zabbix.
Fonte: Autoria própria.

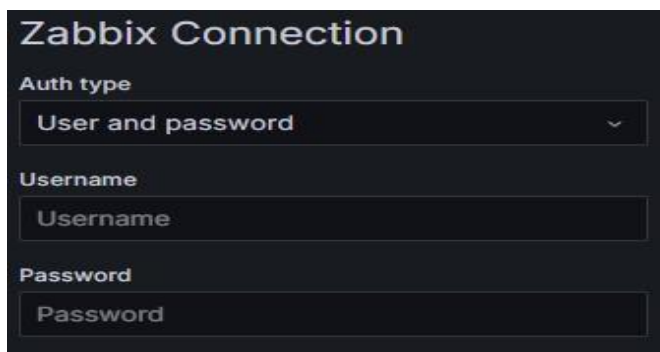


Fig. 17. Tela de configuração da integração do Grafana com o Zabbix.
Fonte: Autoria própria.

Após a inserção dos dados obrigatórios, o administrador deverá clicar no ícone azul “salvar e testar” para validar se a conexão foi realizada com sucesso ou não. Se ocorrer algum erro será mostrado na página a identificação do erro e se for aceito vai aparecer a mensagem verde com a versão do Zabbix integrado, conforme é ilustrado na Fig. 18.

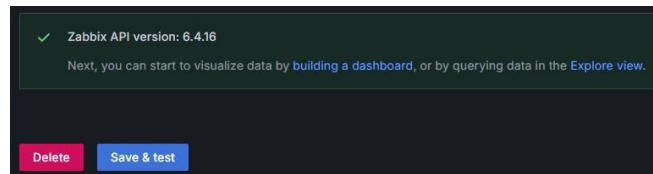


Fig. 18. Tela do resultado da integração do Grafana com o Zabbix.
Fonte: Autoria própria.

Com os *softwares* integrados e trabalhando em sinergia é possível utilizar todas as ferramentas que o Grafana disponibiliza para o desenvolvimento de dashboards personalizados contendo todas as informações que o administrador necessitar e de forma intuitiva.

A Fig. 19 ilustra o dashboard oriundo da integração Grafana/Zabbix no monitoramento dos dispositivos Iots de energia elétrica da EMC.



Fig. 19. Dashboard do Grafana.
Fonte: Autoria própria.

F. Telegram

O Telegram é um aplicativo mensageiro baseado em nuvem onde seu foco principal é a segurança e a velocidade. Seus criadores são Pavel e Nikolai Durov e no seu plano piloto o aplicativo deveria ser super-rápido, simples e gratuito [19].

Atualmente a ferramenta pode ser usada via móvel(celular) ou pelo computador, além de permitir aos usuários enviarem mensagens de texto, voz, vídeos e arquivos de qualquer tipo. Entre essas funções destacam-se:

- **Privacidade, segurança e criptografia:** O aplicativo disponibiliza criação de chats secretos, contas anônimas e ocultação de números, todos esses recursos são criptografados de ponta a ponta assegurando assim a privacidade das comunicações, sem contar que disponibiliza a autodestruição de mensagens.
- **Multiplataforma:** Atualmente o Telegram pode ser usado em vários sistemas como Android, iOS, Windows, Linux dentre outros.
- **Bots e API:** A ferramenta disponibiliza uma API aberta, que através dela se torna possível automatizar ferramentas e integrar sistemas de alerta, sem contar que os usuários podem criar Bots que são robôs.
- **Confiabilidade e Rapidez:** Cumprindo à risca seu plano piloto, atualmente o Telegram possui servidores espalhados pelo mundo e esse feito contribui para a estabilidade e rapidez da ferramenta.

- **Recursos ilimitados e personalizável:** O aplicativo permite criação de grupos com até 200.000 pessoas e canais para enviar mensagens a um público ilimitado, além de proporcionar ao usuário personalizar todo o Frontend como temas, stickers e Gifs animados.
- **Armazenado na Nuvem:** A ferramenta conta com um recurso que o usuário pode acessar de diferentes dispositivos sem haver perda de mensagens e esse recurso só é possível por conta do armazenamento na nuvem.

1) Instalação do Telegram

A instalação do Telegram é feita no celular por conta do software ser um aplicativo de mensagem, logo cada usuário deverá fazer a instalação de forma individual. Com esse requisito concluído o próximo passo é a criação do Bot para posterior prosseguir com a integração com o Zabbix conforme o tutorial de instalação a seguir:

Passo 01: Criação do Bot.

Após a instalação do Telegram, ou caso já possua a ferramenta, é preciso ir ao ícone da lupa e pesquisa BotFather. BotFather é uma ferramenta oriunda do pacote Telegram que é responsável pela criação de todos os bot.

A primeira vez que o BotFather é usado é preciso inserir o comando /start para ele ser inicializado, logo após será recebido uma mensagem informando todos os comandos que podem ser feitos e se faz necessário executar os seguintes comandos:

1. Inserir o comando `/newbot`
2. É solicitado a definição do name do Bot
3. É solicitado a definição do username do Bot.

Obs.: Vale lembrar que o name é username sempre deverão ser diferentes.

Inserido e sendo aceito o name e o username, o BotFather vai retornar uma mensagem informado que o Bot foi criado, juntamente com um token que através dele se torna possível a integração do Telegram com qualquer plataforma que é possível uma comunicação como é o caso do Zabbix conforme é ilustrado na Fig. 20.

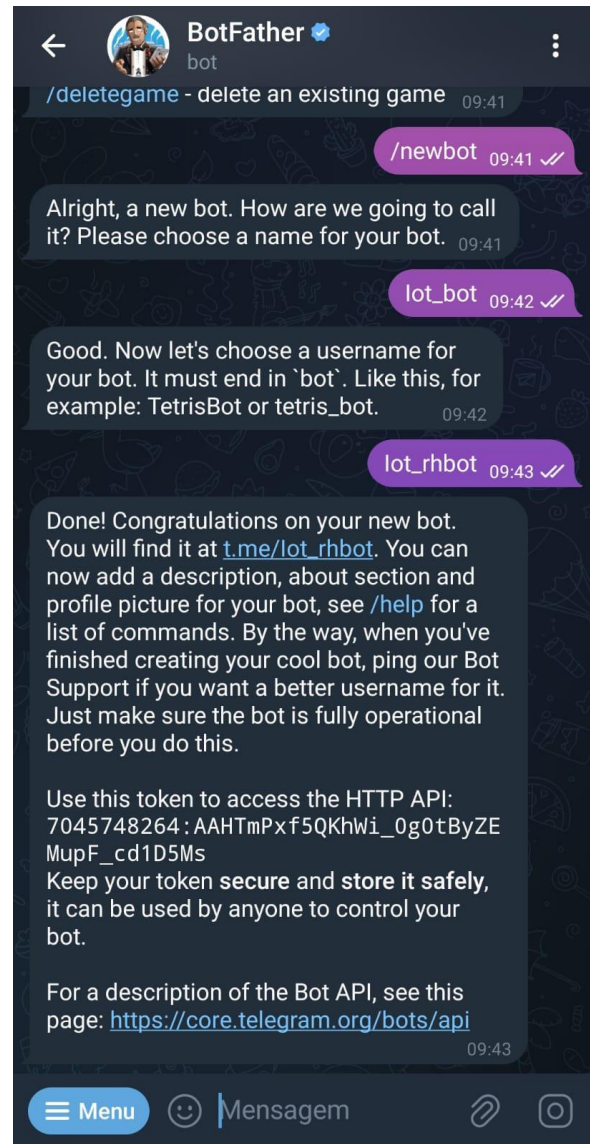


Fig. 20. Tela de criação do Bot.

Fonte: Autoria própria.

O Telegram disponibiliza 2 (duas) opções para o recebimento das mensagens, individual para o criador do bot ou para um grupo de usuários. Logo nos passos a seguir serão discriminado a forma da configuração dos endereços do Bot para o envio das mensagens conforme a demanda que será utilizada.

Passo 02: Configurar para apenas o criador do Bot receber mensagens.

Caso a demanda seja apenas o criador do Bot receber a mensagem, segue-se o mesmo princípio do BotFather, utilizado o IDBot desta vez, primeiro é necessário pesquisa nos contatos o IDBot, e a primeira vez que o IDBot é usado a ferramenta informa que só existem dois comandos válidos `/getid` e `/getgroup`, digitando `/getid` é retornado o ID do Bot conforme a Fig.21 está ilustrando. O uso do ID será discriminado a seguir no momento da integração no Zabbix.

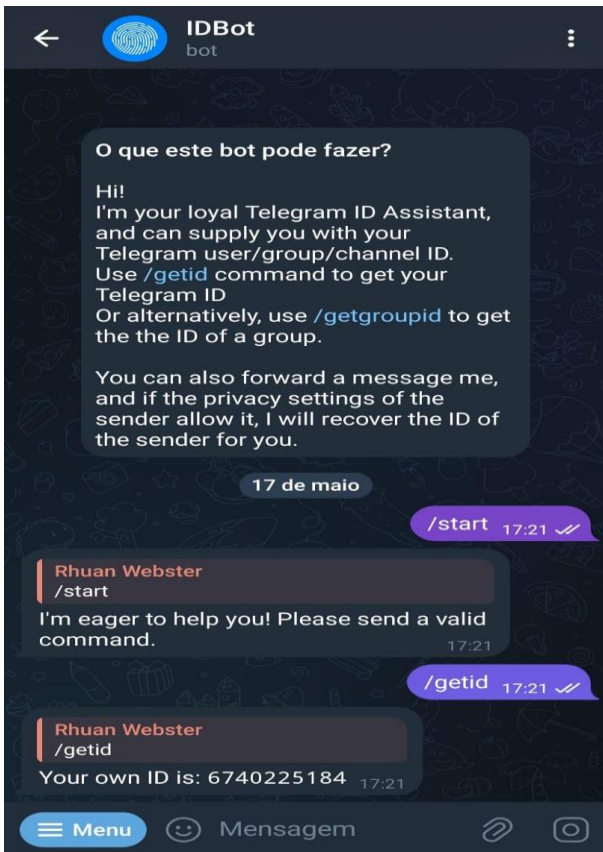


Fig. 21. Tela de criação do ID do Bot.
Fonte: Autoria própria.

Passo 03: Configurar para a mensagem chegar em um grupo.

Caso a demanda seja a mensagem chegar em um grupo, a configuração no Telegram requer apenas o id do grupo para onde o Bot criado está inserido. Logo, primeiramente cria-se o grupo, adiciona o Bot e o IDBot. Então no grupo deverá inserir o comando `/getgroupid@myidbot`, para o IDBot retornar o ID do Supergroup, conforme mostrado na Fig. 22.



Fig. 22. Tela de criação do ID supergroup.
Fonte: Autoria própria.

2) Integração do Telegram com o Zabbix

No Zabbix, a configuração para se executar a integração com o Telegram é relativamente intuitiva, mas necessita seguir os documentos oficiais para não acontecer erros.

A configuração se inicia acessando o menu e selecionando a opção “mídia types”, logo após é preciso configura os parâmetros obrigatórios conforme está sendo ilustrado na Fig. 23 destaque para os campos “TO” e “Token” que deverão ser são preenchidos com o ID do Bot ou o SupergroupID e o Token do Bot respectivamente. Por fim clica-se em update para salva a configuração.

Logo após parte-se para a aba *Message Templates*, nesse modulo é feito a configuração textual das mensagens que serão enviadas, por padrão já vem com uma configuração nativa, mas que poderá ser personalizada e por fim clica-se em update para salva a configuração.

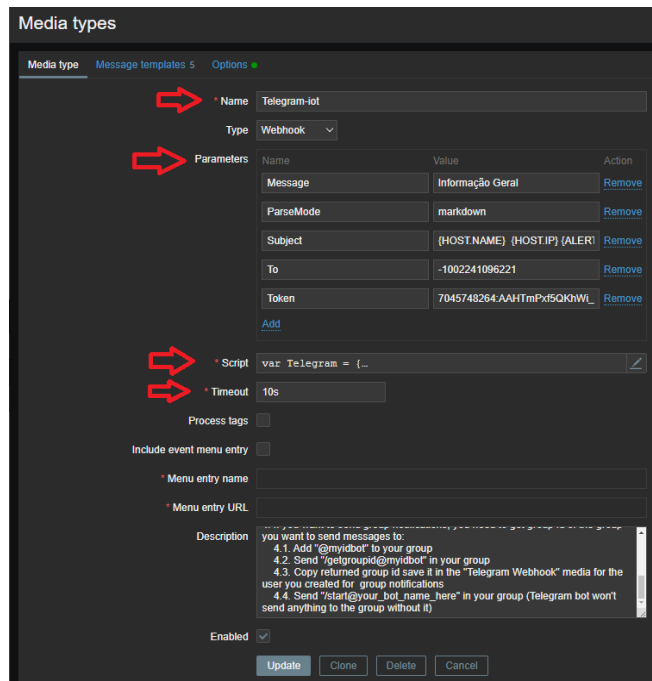


Fig. 23. Tela de integração com configuração dos alertas.
Fonte: Autoria própria.

O próximo passo é habilitar a mídia no usuário do Zabbix, para executar deve-se retorna ao menu do Zabbix, selecionar a opção usuário, clicar no usuário que será habilitado o alerta, clicar na aba mídia e logo após *Add*. O primeiro campo de configuração é o tipo da mídia, deve-se escolher a que foi configurada anteriormente e no campo *send to* informa novamente o id do Bot ou do grupo que irá receber os alertas, por fim deve-se clicar em *add* para salva a configuração.

A Fig. 24 está ilustrando o recurso sendo executado nesse projeto e retornando um problema onde o dispositivo não estava respondendo ao ICMP ping e após a resolução do problema a ferramenta informa que foi resolvido.

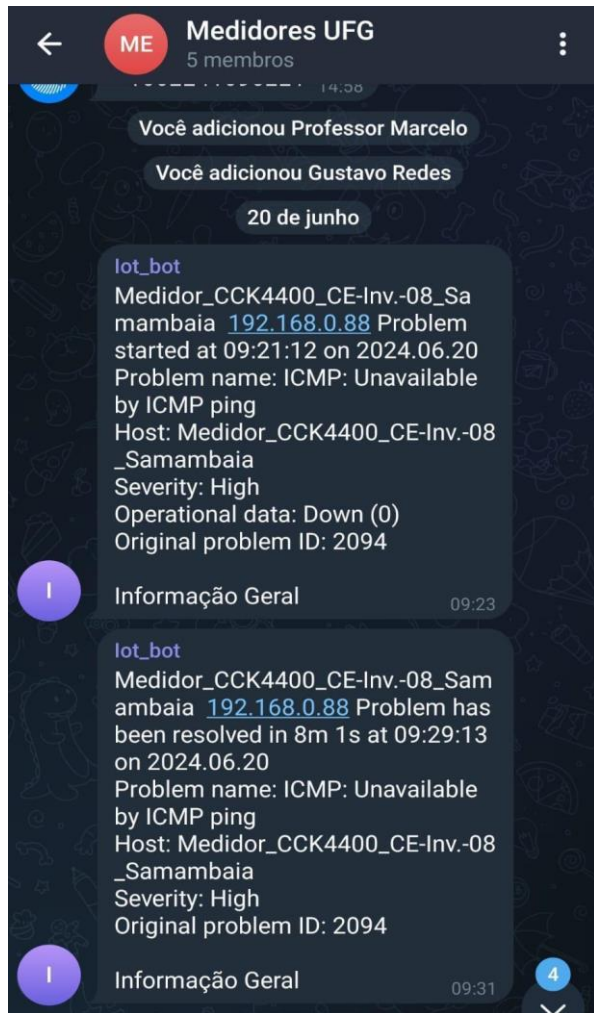


Fig. 24. Tela de resultado da integração reportando alerta.

VI. TESTES E RESULTADOS

As ferramentas Zabbix e Grafana foram instaladas em um servidor na Escola de Engenharia Elétrica, Mecânica e de Computação EMC-UFG, pois dessa forma a solução vai possibilitar acompanhamento e manutenção no futuro por parte da universidade.

O processo de instalação do Zabbix levou 1 (um) dia para ser concluído e o maior desafio ainda estava por vir, que era a forma de comunicação dos dispositivos com o Zabbix.

Logo constatou-se que os dispositivos IoTs existentes possuíam sistema proprietário onde a forma de comunicação Modbus TCP-IP era fechada, inviabilizando assim a comunicação direta.

A solução encontrada foi a consulta ao gerenciador de rede da EMC-UFG para descobrir se havia alguma API que pudesse liberar a comunicação direta e na devolutiva foi apresentado uma API da CCK que trazia dados bem limitados que na sua essência eram leituras de potências ativas e reativas.

Como precisava-se obter o *status* dos dispositivos, logo se pensou que através dos sistemas proprietários era possível identificar o endereço IP de cada equipamento na rede.

Através dele foi visto uma forma de monitorar o status de operação dos medidores e inversores de energia da UFG uma

vez que realizando o ping do respectivo endereço no ambiente de produção do Zabbix os dispositivos retornavam à informação de estar ativo ou inativo.

Logo em seguida começou a inserção de cada dispositivo em forma de host no Zabbix e no cadastro foi realizado o preenchimento do nome, o *template* ICMP Ping que é responsável por ficar monitorando em tempo real toda a resposta do dispositivo na rede, o endereço IP e, pôr fim, a inserção das coordenadas geográficas pois acredita-se ser útil para uma possível manutenção em loco.

Feito o cadastro de todos os dispositivos, começaram os testes dentro do período de 1(um) mês e observou-se requisitos de integridade e confiabilidade das informações. O resultado foi bastante positivo uma vez que o Zabbix conseguiu realizar o monitoramento em tempo real de cada dispositivo sem dificuldade, interrupções e de forma bem simples e confiável.

A próxima fase de um monitoramento é encontrar uma forma de expor esses dados coletados de forma gráfica e que pudesse tornar as informações mais fácil de serem analisadas e compreendidas.

O Zabbix conta com alguns recursos, como por exemplo a exposição de gráficos de cada item de monitoramento, o dashboard inicial que é editável na forma que o usuário desejar, mas foi constatado que era preciso verificar se houvesse uma forma de sofisticada a visualização e foi nesse momento que através de pesquisa descobriu-se o Grafana que é altamente integrável com o Zabbix.

Então foi realizada a instalação do Grafana no mesmo ambiente de produção do Zabbix, esse processo levou 4 horas e foi preciso utilizar a documentação oficial para realizar toda a configuração de integração.

Com as ferramentas em sinergia, foi conduzido testes ao longo de 15 dias focado no desenvolvimento de diferentes dashboards de visualização. Após o período conclui-se que que o Grafana trouxe não apenas facilidade na análise e compreensão dos dados, mas também proporcionou uma curadoria de informações relevantes para o monitoramento. Além disso, o Grafana permitiu o ajuste temporal da exibição dessas informações, otimizando a visualização conforme as necessidades do projeto.

E por fim, para completar o monitoramento é preciso elaborar um sistema de alerta, onde os problemas que aparecerem o sistema retorna avisos para o administrador. O Zabbix possui um módulo de alertas nativo onde os avisos são reportados no dashboard, mas a ferramenta também possui integração com aplicativos de mensagens como por exemplo o Telegram.

Atualmente a população mundial fica muito tempo nesses aplicativos de mensagem uma vez que eles entregam as mensagens/informações de forma bem rápida e no momento que vão acontecendo. Esse requisito é justamente o que o sistema de alerta precisa, então utilizando as documentações oficial do Zabbix foi realizado a integração Zabbix/Telegram.

A instalação e a configuração levaram 1(uma) hora e já seguiu para teste onde o resultado foi bastante satisfatório uma vez que o Telegram informa o problema e quando o mesmo for resolvido é recebido a mensagem também.

TABELA III
TABELA DE DIFICULDADE NA IMPLEMENTAÇÃO DO PROJETO

Métricas	Zabbix	Grafana	Telegram
Dificuldade na instalação	Média	Média	Fácil
Necessidade de pacote adicional	Nenhum	Um Plugin	Nenhum
Tempo de instalação	24 Hrs	4 Hrs	1 Hr
Auto Descoberta de Host	Não	Sim	Não
Dificuldade na configuração	Média	Média	Fácil
Dificuldade na Integração	Média	Média	Média
Qualidade em recursos gráficos	Médio	Grande	Não possui
Facilidade na integração	Médio	Médio	Médio
Qualidade nos recursos de alertas	Baixa	Média	Grande
Mapa e diagramas da rede	Média	Alta	Não Possui
Geração de Gráficos	Média	Alta	Não Possui
Notificações e Alertas	Média	Média	Alta

Fonte: Autoria própria

VII. CONCLUSÕES

No estudo foi proposto realizar a análise de Sistemas de Monitoramento de tráfego em redes de comunicação de dados citados acima, desta forma, pesquisas futuras nesta área são necessárias para garantir um aperfeiçoamento do sistema.

O projeto colocou em funcionamento um sistema de monitoramento estudado com a escolha do Zabbix e Grafana simplificaram a forma de monitorar toda a estrutura de rede dos dispositivos de energia elétrica da UFG.

A implementação das soluções na mesma rede que contém os medidores possibilitou uma facilidade em entender todo o processo de instalação e configuração, sem contar que visualizar a forma que se acontece as coletas de dados abre um horizonte de poder aplicar a mesma solução em ambientes domésticos ou até empresarial.

O Zabbix com a característica de ser gratuito e completo abriu um horizonte para a saída do domínio das fabricantes, pois foi através dos recursos de monitoramento SNMP Ping que viabilizou ser possível realizar o cadastro de cada equipamento, identificá-los com seus respectivos endereços IP e assim atingir o objetivo que era monitorar o status de funcionamento.

O Grafana por sua vez trouxe uma sofisticação no tratamento dos dados coletados, através de seus recursos gráficos foi possível construir painéis com mais informações e de mais fácil entendimento ao administrador ou a qualquer pessoa que esteja incumbida de realizar ou acompanhar o monitoramento.

E por fim o Telegram trouxe uma modernização e visualização muito elegante ao sistema de alerta, onde através da integração sugerida pela plataforma Zabbix o usuário ou o administrador vai poder acompanhar via mobile ou desktop todos os problemas e resolução de problemas dos medidores em tempo real.

O maior desafio encontrado no projeto se mostrou na forma de comunicação com os dispositivos IoTs pois embora eles possuem suporte para o protocolo Modbus TCP IP, suas fabricantes fecharam esses protocolos para apenas o sistema de fábrica obtivessem os dados mais completos, nos deixando apenas com uma API mais simples onde há possibilidade de obter apenas a potência dos dispositivos.

Essa prática é bastante comum entre as fabricantes de equipamentos elétricos pois é dessa forma que os grandes players do mercado *Smart Grid* dominam.

Pensando em trabalhos futuros, viabilizar junto a fabricante a abertura do acesso direto aos dispositivos irá completar a coleta dos dados uma vez que irá ocasionar absorver mais variáveis e sofisticar o monitoramento, caso não seja possível é preciso desenvolver outro modo de obtenção de dados dos dispositivos.

Diante de tudo que foi colocado em prática e testado, comprovou-se que a eficiência dos processos Zabbix, juntamente com a visualização gráfica do Grafana são peças fundamentais para se obter um excelente monitoramento. As soluções trabalhando em sinergia reúnem todos os requisitos essenciais para a coleta de dados, mesmo que embora os outros sistemas estudados tenham as mesmas características, mas na sua grande maioria faz se necessário acoplar licenças empresariais que no caso vai ocasionar custos.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] “MME-Ministerio de Minas e Energia, Balanco energetico nacional,” [Online]. Available: <https://www.gov.br/mme/pt-br/assuntos/secretarias/sntep/publicacoes/balanco-energetico-nacional>. [Acesso em 10 Janeiro 24].
- [2] “PL 624/23,” [Online]. Available: <https://www.camara.leg.br/radio/programas/1059945-camara-aprova-instalacao-de-geradores-de-energia-solar-para-familias-carentes/>. [Acesso em 10 Janeiro 24].
- [3] “Proad, Usinas fotovoltaicas vão gerar 36% da eletricidade consumida pela UFG,” [Online]. Available: <https://proad.ufg.br/n/155341-usinas-fotovoltaicas-va-gerar-36-da-eletricidade-consumida-pela-ufg>. [Acesso em 10 Janeiro 24].
- [4] “Secom-UFG, UFG debate sobre otimização das fontes de energia elétrica,” [Online]. Available: <https://jornal.ufg.br/n/139611-ufg-debate-sobre-otimizacao-das-fontes-de-energia-eletrica>. [Acesso em 10 Janeiro 24].
- [5] J. F. Kurose e K. W. Ross, *Redes de Computadores e a Internet – Uma abordagem Top-Down*, São Paulo: Pearson Education do Brasil, 2013.
- [6] I. C. Peixinho, F. M. d. Fonseca e F. M. Lima, *Segurança de Redes e Sistemas*, Rio de Janeiro: Escola Superior de Redes, 2019.

- [7] “ABB, monitoramento de corrente,” [Online]. Available: <https://loja.br.abb.com/blog/post/monitoramento-de-corrente>. [Acesso em 10 Janeiro 24].
- [8] “Checkmk,” [Online]. Available: <https://checkmk.com/>. [Acesso em 10 Janeiro 24].
- [9] “CloudStats,” [Online]. Available: <https://servermonitoring.me/br/>. [Acesso em 10 Janeiro 24].
- [10] “Zabbix,” [Online]. Available: <https://www.zabbix.com/documentation/6.4/en/manual>. [Acesso em 30 Janeiro 2024].
- [11] “Spiceworks,” [Online]. Available: <https://www.spiceworks.com/>. [Acesso em 10 Janeiro 24].
- [12] “Splunk,” [Online]. Available: <https://www.splunk.com/>. [Acesso em 10 Janeiro 24].
- [13] “CCK,” [Online]. Available: <https://cckautomacao.com.br/produtos/>. [Acesso em 20 Janeiro 2024].
- [14] “Huawei,” [Online]. Available: <https://support.huawei.com/enterprise/br/doc/EDOC1100024501>. [Acesso em 20 Janeiro 2024].
- [15] “WEG,” [Online]. Available: <https://www.weg.net/institucional/BR/pt/digital-solutions/solutions/gateways>. [Acesso em 20 Janeiro 24].
- [16] “Instalação Zabbix e MariaDB,” [Online]. Available: <https://technologyrss.com/how-to-install-zabbix-6-4-on-debian-12/>. [Acesso em 10 Janeiro 24].
- [17] “APACHE,” [Online]. Available: <https://www.hostinger.com.br/tutoriais/o-que-e-apache>. [Acesso em 25 Janeiro 24].
- [18] “Grafana,” [Online]. Available: <https://grafana.com/>. [Acesso em 20 Março 2024].
- [19] “Telegram,” [Online]. Available: <https://telegram.org/>. [Acesso em 20 Abril 2024].



Rachimi Solarevisky de Jesus, estudante de graduação do curso de Engenharia de Computação na Universidade Federal de Goiás – UFG. Possui experiência na área de Suporte Técnico em ambiente Empresarial, Servidor Windows/Linux. Atuando como CEO de uma Startup de Impressão 3D, desde 2019.



Marcelo Stehling de Castro graduou-se em Engenharia Elétrica pela Universidade Federal de Juiz de Fora (1992), com mestrado em Engenharia Elétrica pela Universidade Estadual de Campinas (1995) e doutorado em Engenharia Elétrica pela UnB (2010). Professor Associado da Universidade Federal de Goiás, tendo ingressado em 1996. Possui experiência na área de engenharia de redes, computação paralela e distribuída, comunicações óticas e tecnologias alternativas de última milha (BPL, ZigBee, Wi-Fi). Desenvolve pesquisas em temas que incluem redes de comunicação (5G, Gigabit Wi-Fi), *Smart Grids*, *Smart Cities*, *Smart Campus*, tecnologia da informação e comunicação e gestão aplicadas a projetos de redes de telecomunicações, projetos de automação usando Plataforma Arduino e educação em engenharia.



Rhuan Webster de Lourenço e Silva, estudante de graduação do curso de Engenharia de Computação na Universidade Federal de Goiás - UFG. Possui experiência na área de suporte técnico, atuando como Suporte Técnico Sênior na Agência Goiana de Defesa Agropecuária – AGRODEFESA-GO desde 2011.