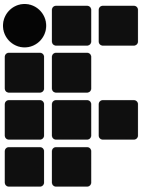


INSTITUTO FEDERAL
Farroupilha
Campus Uruguaiana

CURSO TÉCNICO EM INFORMÁTICA INTEGRADO AO ENSINO MÉDIO

PEDRO GONZALEZ

SISTEMA DE MONITORAMENTO DE REDES

URUGUAIANA

2025

PEDRO GONZALEZ

SISTEMA DE MONITORAMENTO DE REDES

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Informática Integrado ao Ensino Médio do *Campus* Uruguaiana do Instituto Federal de Educação, Ciência e Tecnologia Farroupilha como requisito parcial para a obtenção do título de Técnico em Informática.

Orientadores:

João Carlos de Carvalho e Silva Ribeiro

Pablo Brauner Viegas

URUGUAIANA

2025

Gonzalez, Pedro.
Sistema de monitoramento de redes / Pedro Gonzalez. – 2025.

45 f.

Trabalho de Conclusão de Curso Técnico – Instituto Federal de Educação, Ciência e Tecnologia Farroupilha, Uruguaiana, 2023.

1. Monitoramento. 2. Alertas automatizados. 3. Ferramentas. I. Sistema de monitoramento de redes.

CDD [número da CDD].

PEDRO GONZALEZ

SISTEMA DE MONITORAMENTO DE REDES

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Informática Integrado ao Ensino Médio do *Campus* Uruguaiana do Instituto Federal de Educação, Ciência e Tecnologia Farroupilha como requisito parcial para a obtenção do título de Técnico em Informática.

Este trabalho foi defendido e aprovado pela banca em 19/12/2025.

BANCA EXAMINADORA

_____
Prof. Ms. João Carlos de Carvalho e Silva Ribeiro
Orientador

_____
Prof. Ms. Pablo Brauner Viegas
Orientador

_____
Prof. Ms. Jonathan Alberto dos Santos Silveira
Avaliador

_____
Prof. Ms. Anderson Mendes Rocha
Avaliador

Dedico este trabalho à minha família, orientadores, colegas, amigos e ao Instituto Federal Farroupilha – Campus Uruguaiana, por todo o apoio, incentivo e contribuição ao longo desta jornada acadêmica.

AGRADECIMENTOS

Dedico este trabalho ao Instituto Federal Farroupilha – Campus Uruguaiana, em reconhecimento aos três anos de formação que vivencio na instituição, os quais foram fundamentais para meu crescimento acadêmico e pessoal.

Manifesto minha sincera gratidão, em primeiro lugar, à minha família, Ana Paula Gonzalez, Gleci da Silva Gonzalez e Paulo Sérgio Stautmaster Gonzalez pelo apoio incondicional, pela presença constante nos momentos mais difíceis e pelo incentivo durante todas as etapas do desenvolvimento deste trabalho.

Agradeço, de maneira especial, aos meus orientadores, professores João Ribeiro e Pablo Viegas, pelo comprometimento, pela orientação criteriosa desde a definição do tema até a proposta de aplicação futura em ambiente de produção, bem como pela disponibilidade contínua para sanar dúvidas e oferecer direcionamento técnico e metodológico.

Registro também meu profundo agradecimento à minha namorada, Anna Brasil, pela paciência, incentivo e apoio emocional, que foram essenciais para que eu me mantivesse sereno e motivado diante dos desafios enfrentados durante a elaboração deste trabalho.

Aos colegas e amigos do campus, em especial Daniel Dorneles, Pedro Rossa e Samuel Prates, agradeço pelo companheirismo, pelas palavras de incentivo e pela colaboração nos momentos de maior necessidade. Fora do ambiente acadêmico, expresso minha gratidão aos amigos João Fortes, Vinicios Guedes e Jonatas Gomes, por compartilharem momentos de lazer, apoio emocional e trocas de ideias que contribuíram significativamente para o amadurecimento do projeto.

Por fim, agradeço a todas as pessoas que, direta ou indiretamente, contribuíram para a realização deste trabalho. Mesmo que não mencionadas nominalmente, deixo aqui meu reconhecimento e apreço por sua colaboração em minha trajetória acadêmica.

Nós não podemos mudar o que está feito, podemos apenas seguir em frente.

ARTHUR MORGAN

RESUMO

Este Trabalho de Conclusão de Curso apresenta o desenvolvimento e a validação, em ambiente virtualizado, de um sistema de monitoramento de redes com alertas automatizados, voltado ao Instituto Federal Farroupilha – Campus Uruguaiana. A proposta fundamenta-se na necessidade institucional de aprimorar a supervisão da infraestrutura de rede, reduzindo o tempo de detecção e resposta a falhas que impactam atividades pedagógicas e administrativas. A solução baseia-se na integração de ferramentas de código aberto, utilizando o Zabbix para coleta e gerenciamento de métricas, o Grafana para visualização gráfica e análise em tempo real, e o Telegram como canal de envio automatizado de alertas por meio de bots. A metodologia adotada caracteriza-se como pesquisa aplicada, de natureza exploratória e abordagem qualitativa, contemplando revisão bibliográfica, estudo técnico das ferramentas, configuração de ambiente virtualizado, integração dos sistemas e execução de testes simulando falhas e degradações de serviço. Os resultados demonstram a viabilidade técnica da solução proposta, evidenciando estabilidade na coleta de dados, clareza na visualização das métricas e eficiência no envio de notificações em tempo real. Conclui-se que o sistema desenvolvido apresenta potencial para futura implantação em ambiente de produção, contribuindo para uma gestão de redes mais proativa, eficiente e economicamente viável em instituições públicas de ensino.

Palavras-chave: Monitoramento de redes; Zabbix; Grafana; Telegram; Alertas automatizados.

ABSTRACT

This Final Course Project presents the development and validation, in a virtualized environment, of a network monitoring system with automated alerts, aimed at the Instituto Federal Farroupilha – Uruguaiana Campus. The proposal is grounded in the institutional need to enhance network infrastructure supervision, reducing the time required to detect and respond to failures that affect pedagogical and administrative activities. The solution is based on the integration of open-source tools, using Zabbix for metric collection and management, Grafana for graphical visualization and real-time analysis, and Telegram as an automated alert delivery channel through bots. The adopted methodology is classified as applied research, exploratory in nature, with a qualitative approach, encompassing a literature review, technical analysis of the selected tools, configuration of a virtualized environment, system integration, and execution of tests simulating service failures and performance degradation. The results demonstrate the technical feasibility of the proposed solution, highlighting stability in data collection, clarity in metric visualization, and efficiency in real-time alert notifications. It is concluded that the developed system shows strong potential for future deployment in a production environment, contributing to a more proactive, efficient, and cost-effective network management model in public educational institutions.

Keywords: Network Monitoring; Zabbix; Grafana; Telegram; Automated Alerts.

LISTA DE ILUSTRAÇÕES

LISTA DE ABREVIATURAS E SIGLAS

API	Application Programming Interface
CPU	Central Processing Unit
GID	Group Identifier
GPU	Graphics Processing Unit
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
ICMP	Perda de Pacotes por ICMP
IFFar	Instituto Federal Farroupilha
ID	Identificador
IP	Internet Protocol
LTS	Long Term Support
MySQL	My Structured Query Language
PING	Packet Internet Network Grouper
RAM	Random Access Memory
SMTP	Simple Mail Transfer Protocol
SSH	Secure Shell
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol / Internet Protocol
TLS	Transport Layer Security
URL	Uniform Resource Locator
VM	Virtual Machine

SUMÁRIO

1 INTRODUÇÃO

A sociedade contemporânea é marcada por uma profunda transformação digital, em que o uso intensivo da tecnologia se tornou essencial para o funcionamento de instituições em praticamente todos os setores. No contexto da educação, essa dependência é ainda mais acentuada: escolas, universidades e institutos federais utilizam tecnologias digitais para apoiar e viabilizar atividades administrativas, pedagógicas e comunicacionais. Tarefas como matrícula, acesso a sistemas acadêmicos, organização de aulas, armazenamento de documentos, videoconferências e envio de comunicados são, em grande parte, realizadas por meio da infraestrutura de redes de computadores.

Uma rede de computadores pode ser compreendida como um conjunto de dispositivos conectados entre si – como computadores, celulares, impressoras, roteadores e servidores – que trocam informações de forma organizada. Essa rede pode funcionar de forma local, dentro da própria instituição, ou conectada à internet, permitindo o acesso a recursos externos. Para que tudo funcione adequadamente, é necessário que essa rede opere com estabilidade, segurança e desempenho satisfatório.

No entanto, falhas podem ocorrer por diversos motivos: defeitos em equipamentos, quedas de energia, congestionamentos, ataques virtuais ou erros de configuração. Quando não são identificadas rapidamente, essas falhas podem prejudicar seriamente o funcionamento da instituição, afetando desde aulas e atendimentos até o acesso a sistemas administrativos. Muitas vezes, a equipe responsável pela tecnologia só é informada da falha após o problema já ter causado prejuízos, o que dificulta uma resposta ágil e eficaz.

Para enfrentar esse desafio, utiliza-se o monitoramento de redes, que consiste no uso de programas específicos para acompanhar o funcionamento dos dispositivos da rede em tempo real. Um dos principais softwares utilizados nesse tipo de tarefa é o Zabbix, que é gratuito e de código aberto – ou seja, sem custos de licenciamento e com possibilidade de ser adaptado às necessidades da instituição. O Zabbix permite verificar se os equipamentos estão ligados, se estão funcionando corretamente, se há sobrecarga ou falhas de comunicação, entre outros indicadores importantes.

Há um sistema que já está em uso no Instituto Federal Farroupilha – Campus Uruguaiana, o pfSense, que atua como um *firewall* – um tipo de “porteiro digital” que filtra o que pode ou não entrar e sair da rede, protegendo contra acessos indevidos, tentativas de invasão e uso inadequado da internet.

Contudo, mesmo com essas ferramentas em operação, ainda há limitações importantes. Atualmente, não há um mecanismo automatizado que envie alertas imediatamente aos técnicos quando uma falha ocorre. Isso significa que, mesmo quando é identificado um problema, é necessário que alguém observe manualmente os dados para tomar providências. Assim, muitas falhas passam despercebidas por algum tempo, até que usuários relatem o erro, o que pode comprometer a continuidade das atividades da instituição.

Para solucionar esse problema, este Trabalho de Conclusão de Curso propôs o desenvolvimento de um sistema automatizado de envio de alertas, integrando o Zabbix com o aplicativo de mensagens Telegram através do Grafana. O Telegram é uma plataforma gratuita de troca de mensagens entre pessoas, que permite a criação de *bots* – pequenos programas automatizados capazes de enviar mensagens de forma inteligente. A ideia central é que, ao detectar uma falha ou situação fora do normal, o sistema envie imediatamente uma mensagem para o celular da equipe técnica, permitindo que o problema seja tratado com rapidez, muitas vezes antes mesmo de ser percebido pelos usuários finais.

Além do envio de alertas, o projeto também visou implementar dashboards visuais interativos para facilitar a visualização do estado da rede em tempo real. Para isso, será utilizada a ferramenta Grafana, que também é gratuita e de código aberto. O Grafana permite criar painéis gráficos que organizam e exibem os dados coletados pelo Zabbix de maneira clara, dinâmica e visualmente compreensível. Com esses painéis, os técnicos poderão acompanhar, por exemplo, quais servidores estão ativos, qual o nível de uso de memória e processamento dos equipamentos, quais serviços estão com desempenho abaixo do esperado e outros indicadores relevantes, além de possuir integrações diretas com outras ferramentas, como o Telegram. Isso torna o processo de monitoramento mais intuitivo, acessível e eficiente.

A construção e os testes iniciais do sistema foram realizados em um ambiente virtualizado, utilizando o programa Oracle VirtualBox, que permite criar simulações de redes dentro do próprio computador. Isso possibilitou experimentar e aperfeiçoar

o sistema sem colocar em risco a operação da rede real da instituição. Após a validação do funcionamento e da segurança da solução no ambiente simulado, está prevista sua implantação na infraestrutura do IFFar – Campus Uruguaiana.

Cabe destacar que a presente proposta foi concebida considerando o contexto de restrições orçamentárias enfrentado pelas instituições públicas de ensino superior, as quais, de modo recorrente, dispõem de recursos financeiros limitados para a aquisição e manutenção de softwares comerciais de alto custo. Conforme manifestações oficiais da Associação Nacional dos Dirigentes das Instituições Federais de Ensino Superior (Andifes), a insuficiência do orçamento discricionário tem comprometido o funcionamento regular das universidades federais, impactando diretamente áreas estratégicas, incluindo infraestrutura tecnológica e inovação. Nesse cenário, a adoção de ferramentas livres, gratuitas e personalizáveis apresenta-se como uma alternativa técnica e economicamente viável, capaz de atender às demandas institucionais sem ampliar os custos operacionais.

Assim, o objetivo geral deste trabalho é desenvolver e validar um sistema de monitoramento de redes para o IFFar – Campus Uruguaiana, utilizando tecnologias de código aberto e canais de comunicação acessíveis, de modo a melhorar a capacidade institucional de supervisão e gestão da infraestrutura de rede.

1.1 JUSTIFICATIVA

A adoção de tecnologias de código aberto tem se consolidado como uma estratégia eficaz para o fortalecimento da infraestrutura de Tecnologia da Informação (TI) em instituições públicas de ensino, permitindo maior autonomia, redução de custos e flexibilidade para adaptações conforme as necessidades específicas de cada ambiente institucional. Estudos recentes demonstram que ferramentas como o Zabbix e o Grafana, amplamente difundidas no contexto educacional, são capazes de promover monitoramento eficiente e visualmente acessível de redes e serviços, mesmo em ambientes com recursos financeiros limitados (Souza, Hanna e Acosta, 2020; Santos, 2011).

No âmbito do Instituto Federal Farroupilha – Campus Uruguaiana, o uso de soluções consolidadas como o pfSense, como *firewall*, já representa um avanço significativo na estrutura de TI. Contudo, ainda se observa uma lacuna crítica: a

ausência de mecanismos de monitoramento eficazes e notificações automatizadas que alertem os técnicos responsáveis assim que falhas são identificadas. Essa ausência compromete a capacidade de resposta rápida frente a incidentes e pode resultar em longos períodos de indisponibilidade de serviços essenciais.

A relevância deste projeto está, portanto, em propor uma integração estratégica de ferramentas já consolidadas no contexto institucional, com o acréscimo de uma camada de automação comunicacional baseada em alertas instantâneos via Telegram. Diferente de abordagens já implementadas que tratam apenas da configuração isolada do Zabbix para monitoramento de dispositivos (Andreoli, 2016; Sena e Dutra, 2021), este trabalho visou combinar, em um único sistema, o monitoramento técnico do Zabbix, a visualização intuitiva de dados por meio do Grafana e a eficiência da comunicação automatizada via **bot** do Telegram.

Essa proposta representa um sistema funcional no ecossistema de redes institucionais, pois ultrapassa o modelo tradicional de supervisão passiva e promove uma atuação proativa e responsiva por parte da equipe técnica. O envio automático de mensagens em tempo real, com base em gatilhos definidos no Zabbix, reduz drasticamente o tempo entre a ocorrência da falha e sua identificação pelos operadores, como demonstrado empiricamente por Souza, Hanna e Acosta (2020), que ressaltam a eficácia da integração Zabbix-Grafana na redução de tempo de resposta em ambientes educacionais.

A estruturação do sistema em ambiente virtualizado, utilizando o Oracle VirtualBox, permite ainda uma abordagem segura e controlada para a implementação inicial, sem risco de comprometer os serviços reais durante o desenvolvimento. Essa estratégia é especialmente relevante em instituições públicas, que não podem assumir interrupções operacionais durante testes e validações técnicas.

Além disso, a proposta tem potencial de replicabilidade em outras unidades da rede federal de educação, justamente por utilizar exclusivamente ferramentas livres e acessíveis, acompanhadas de documentação técnica. Como destacado por Santos (2011), o uso de softwares livres como o Zabbix permite não apenas reduzir custos, mas também promover o conhecimento técnico interno e a autonomia institucional no gerenciamento da TI.

Portanto, o projeto justifica-se não apenas por suprir uma necessidade concreta e não atendida no contexto do Campus Uruguaiana, mas também por

representar um modelo viável e escalável para o aprimoramento da gestão de redes em instituições públicas de ensino.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Desenvolver e validar, em ambiente virtualizado, um sistema de alertas automatizados para monitoramento de rede, utilizando ferramentas de código aberto, como Zabbix e Grafana integradas ao Telegram, com o objetivo de aprimorar a capacidade de resposta a falhas no Instituto Federal Farroupilha – Campus Uruguaiana e viabilizar sua posterior implantação em ambiente de produção.

1.2.2 Objetivos Específicos

Os objetivos específicos são:

- Instalar e configurar o Zabbix para realizar o monitoramento de dispositivos e serviços de rede e, integrar o Zabbix com o Grafana para a construção de dashboards gráficos interativos e personalizados.
- Desenvolver um bot no Telegram e integrá-lo ao Zabbix via Grafana para envio automatizado de alertas técnicos.
- Validar o funcionamento do sistema por meio da execução de testes simulando falhas e analisando o comportamento do sistema quanto à detecção e resposta a incidentes.
- Documentar todas as etapas de configuração, incluindo descrições e justificativas das ferramentas utilizadas na construção do projeto.

1.3 METODOLOGIA

A presente pesquisa classifica-se como aplicada, de natureza exploratória e com abordagem qualitativa. O foco reside na concepção, desenvolvimento e validação de um sistema de monitoramento de redes, voltado à realidade institucional do IFFar – Campus Uruguaiana. A metodologia adotada segue uma sequência lógica de etapas que visam garantir tanto a fundamentação teórica quanto

a consistência prática da solução desenvolvida. As etapas do trabalho são descritas a seguir:

- **Definição do Tema:** A partir de diálogos iniciais com os orientadores do projeto, foi identificada a necessidade institucional de um sistema capaz de monitorar a infraestrutura de rede e emitir alertas automáticos em caso de falhas, degradação de desempenho ou indisponibilidade de serviços.
- **Pesquisa Bibliográfica:** Realizou-se uma revisão de literatura com foco em trabalhos acadêmicos, artigos científicos e manuais técnicos que tratam da aplicação de ferramentas de monitoramento com Zabbix e Grafana, bem como a utilização de notificações via Telegram.
- **Estudo das Ferramentas:** Foi conduzida uma análise funcional detalhada das ferramentas Zabbix (para coleta e gestão de métricas), Grafana (para visualização de dados) e Telegram (como meio de envio de notificações automáticas). Avaliaram-se aspectos como compatibilidade, requisitos de instalação, configurações recomendadas e suporte da comunidade.
- **Configuração do Ambiente de Testes:** Para fins de experimentação controlada, foi-se criado um ambiente virtualizado com máquinas Linux (Ubuntu Desktop 24.04.2 (64-bit)) utilizando o Oracle VirtualBox. Nesse ambiente, foram instaladas e configuradas todas as ferramentas mencionadas. Essa etapa visou validar previamente a integração e o funcionamento dos componentes, evitando interferência no ambiente de produção institucional.
- **Integração dos Componentes:** Foi estabelecida a comunicação entre as ferramentas, conectando o Zabbix ao Grafana por meio de um plugin não nativo do Grafana, e programando scripts de alerta para acionar um bot do Telegram. Essa fase envolveu a criação de dashboards personalizados no Grafana e o uso de webhooks para automatizar os alertas enviados via Telegram. A integração desses componentes inclui-se dentro do ambiente de testes.
- **Testes e Validação:** Foram executados testes simulando falhas e degradações em dispositivos monitorados, a fim de verificar o comportamento do sistema quanto à detecção de eventos, geração de

alertas e visualização em tempo real. A validação considerou critérios de tempo de resposta, clareza das informações geradas e confiabilidade da comunicação com o Telegram.

- **Elaboração do Trabalho:** Paralelamente às etapas técnicas, procede-se à redação deste TCC, de acordo com as normas da ABNT, sistematizando a fundamentação teórica, a descrição metodológica, os resultados obtidos e as considerações finais.

2 DESENVOLVIMENTO

2.1 REFERENCIAL TEÓRICO

O monitoramento de redes tornou-se um componente estruturante para a garantia de disponibilidade, desempenho e segurança em ambientes computacionais contemporâneos. A expansão da complexidade tecnológica e a crescente dependência de serviços digitais nas instituições de ensino exigem ferramentas capazes de coletar métricas em tempo real, detectar falhas automaticamente e fornecer subsídios à tomada de decisão. Nesse contexto, soluções de código aberto têm ganhado destaque devido à sua flexibilidade, custo reduzido e capacidade de integração.

Entre essas soluções, o Zabbix consolidou-se como uma plataforma amplamente adotada para monitoramento de redes, servidores e serviços. De acordo com Santos (2011), suas características de extensibilidade, robustez e suporte a diversos métodos de coleta o tornam particularmente adequado para instituições públicas com restrições orçamentárias. Estudos mais recentes, como o de Sena e Dutra (2021), reforçam essa visão ao demonstrar a capacidade do Zabbix em monitorar ambientes distribuídos, correlacionar métricas e acionar gatilhos para detecção de anomalias.

Entretanto, apesar da eficiência na coleta e processamento de dados, as visualizações nativas do Zabbix são relativamente limitadas em termos de flexibilidade e personalização. Para suprir essa lacuna, ferramentas complementares como o Grafana têm sido amplamente utilizadas. Souza, Hanna e Acosta (2020) destacam que o uso combinado de Zabbix e Grafana permite a criação de dashboards dinâmicos, facilitando análises mais precisas e uma compreensão imediata do comportamento da rede. O Grafana, ao consumir dados diretamente da API do Zabbix, possibilita representações gráficas otimizadas para observação contínua e detecção visual de tendências e irregularidades.

Além da coleta e visualização, outro elemento fundamental em sistemas modernos de monitoramento é a capacidade de enviar alertas de forma automatizada e imediata. Nesse sentido, ampliam-se as abordagens baseadas em mensageria instantânea, impulsionadas pela disponibilidade de APIs abertas e gratuitas. O Telegram destaca-se como uma alternativa viável justamente por

oferecer uma interface programável, com suporte para bots que podem receber mensagens automatizadas oriundas de sistemas externos. A documentação oficial do Telegram (2025) descreve mecanismos simples e eficientes de integração via *Hypertext Transfer Protocol* (HTTP) e *JavaScript Object Notation* (JSON), que permitem seu uso como canal de alerta de alta disponibilidade, tornando-o compatível com plataformas como o Grafana.

A integração entre Zabbix, Grafana e Telegram configura uma abordagem coerente com os princípios modernos de observabilidade, que envolvem não apenas monitorar, mas também comunicar o estado da infraestrutura de forma ágil e contextualizada. A literatura técnica aponta que sistemas integrados de monitoramento reduzem significativamente o tempo entre a ocorrência da falha e a intervenção operacional, mitigando impactos sobre usuários e serviços críticos (Souza; Hanna; Acosta, 2020).

Além disso, embora ferramentas como o pfSense desempenhem papel relevante na segurança de ambientes institucionais, sua contribuição aqui consiste sobretudo como parte do ecossistema mais amplo de rede, e não como componente direto da implementação testada. A documentação da Netgate (2025) reforça que *firewalls* como o pfSense podem coexistir com plataformas de monitoramento sem interferir no fluxo interno de coleta e análise de métricas, sendo portanto compatíveis com a proposta de futura implantação em ambiente real.

2.1.1 Trabalhos e Sistemas Semelhantes

Diversos estudos têm explorado o uso de ferramentas de código aberto para monitoramento de redes, destacando abordagens que combinam coleta de métricas, visualização avançada e mecanismos de alerta. Essas iniciativas oferecem subsídios importantes para o desenvolvimento de soluções integradas em ambientes institucionais, como o proposto neste trabalho.

Sena e Dutra (2021) apresentaram uma arquitetura de monitoramento baseada no Zabbix aplicada a servidores Linux, demonstrando sua eficiência na detecção de falhas e na coleta contínua de métricas. Contudo, os autores concentraram-se exclusivamente no uso do Zabbix, sem incorporar ferramentas externas de visualização ou mecanismos de notificação instantânea, limitando o alcance da solução quanto à comunicação rápida de eventos críticos.

Souza, Hanna e Acosta (2020) avançaram nesse aspecto ao investigar a integração entre Zabbix e Grafana, enfatizando a criação de dashboards interativos e personalizáveis. O estudo demonstrou a capacidade do Grafana de ampliar a observabilidade ao transformar métricas técnicas em representações visuais intuitivas. Ainda assim, a proposta não inclui canais de alerta por mensageria instantânea, deixando de abordar a etapa essencial de comunicação automática com operadores de rede.

Santos (2011) analisou a adoção de ferramentas livres em instituições de ensino, ressaltando a relevância de sistemas de monitoramento de baixo custo e de fácil manutenção. Seu trabalho evidencia a viabilidade técnica e econômica do uso de plataformas como o Zabbix em contextos públicos. Entretanto, não contempla integrações avançadas nem funcionalidades modernas de automação de alertas.

Em outra direção, Andreoli (2016) investigou a aplicação do Zabbix em infraestruturas de médio porte, destacando a robustez, a escalabilidade e os benefícios operacionais decorrentes da implantação. Apesar disso, o autor não abrangeu mecanismos complementares de visualização ou comunicação automatizada, mantendo foco na operação tradicional do Zabbix.

Além dos trabalhos acadêmicos, outras ferramentas de monitoramento têm sido utilizadas em ambientes corporativos e educacionais. Um exemplo relevante é o Pandora FMS, que oferece funcionalidades comparáveis às do Zabbix, incluindo monitoramento distribuído e coleta de métricas. No entanto, suas versões mais completas dependem de licenciamento pago, reduzindo sua atratividade para instituições públicas que necessitam de soluções integralmente abertas.

A análise desses estudos evidencia que, embora existam propostas consistentes envolvendo o Zabbix e integrações com o Grafana, raramente são abordadas soluções que unifiquem monitoramento, visualização e comunicação automatizada por meio de plataformas de mensagens instantâneas. Dessa forma, o presente trabalho distingue-se ao propor a integração entre Zabbix, Grafana e Telegram, formando um sistema completo que abrange coleta, análise visual e notificações em tempo real.

2.2 FUNDAMENTAÇÃO PRÁTICA

Tendo como objetivo central desenvolver e validar, em ambiente virtualizado, um sistema de alertas automatizados para o monitoramento da infraestrutura de redes do IFFar – Campus Uruguaiana, optou-se por uma abordagem técnico-prática que integra ferramentas relevantes ao contexto do projeto, como o Zabbix, além da utilização estratégica do Telegram e do Grafana como canais de notificação em tempo real. Embora o pfSense tenha sido considerado durante a fase de planejamento, ele não foi empregado na implementação final.

A fundamentação deste projeto está ancorada no entendimento de que o ambiente virtualizado deve representar fielmente a realidade da rede física do campus. Com isso, busca-se garantir que as condições observadas durante os testes sejam compatíveis com aquelas encontradas no ambiente de produção, permitindo que os resultados obtidos na simulação sejam diretamente aplicáveis em uma futura implantação.

A proposta difere de implementações tradicionais de monitoramento por enfatizar a integração funcional entre diversas tecnologias livres, com destaque para a automação de respostas via Telegram e a visualização inteligente de dados por meio de *dashboards* no Grafana. A integração entre esses componentes constitui o núcleo do trabalho, não apenas pela eficiência operacional que proporciona, mas também pela viabilidade técnica e econômica em contextos institucionais com recursos limitados.

Assim, a presente seção descreve, as ferramentas envolvidas e os fundamentos técnicos que embasam sua utilização coordenada no ambiente virtual proposto.

2.2.1 Zabbix

O Zabbix é uma plataforma de monitoramento de redes e serviços baseada em uma arquitetura modular que separa suas funções em componentes específicos, garantindo escalabilidade, desempenho e organização operacional. O Zabbix Server constitui o núcleo do sistema: é o responsável por processar as métricas recebidas dos hosts monitorados, armazená-las no banco de dados e avaliar os gatilhos configurados para detectar incidentes. O Zabbix Agent é instalado nos dispositivos monitorados e tem a função de coletar informações locais, como uso de CPU, memória, disco, disponibilidade de serviços e integridade do sistema operacional,

enviando esses dados ao servidor central. Já o Zabbix Frontend corresponde à interface web da plataforma, desenvolvida em PHP e executada sobre um servidor Apache, possibilitando a visualização de métricas, configuração de hosts, criação de gatilhos, consultas históricas e administração completa do sistema. A interação coordenada entre esses três componentes permite que o Zabbix desempenhe monitoramento contínuo, detecção de falhas e análise de desempenho em ambientes distribuídos e de elevada complexidade.

A instalação da ferramenta foi realizada em uma máquina virtual configurada com Ubuntu Desktop 24.04, utilizando-se a versão Zabbix 7.0 LTS, reconhecida por sua estabilidade, ciclo de suporte prolongado e ampla compatibilidade com servidores web Apache e banco de dados MySQL, adotados neste projeto. A instalação seguiu integralmente as diretrizes oficiais da Zabbix SIA, contemplando a implementação dos componentes Zabbix Server, Frontend e Zabbix Agent, além da criação e configuração do banco de dados em MySQL e da interface web operando sobre Apache.

Com a finalização da instalação, o Zabbix passou a operar por meio de sua arquitetura cliente-servidor, na qual o Zabbix Server realiza o processamento das métricas, o Frontend fornece a interface de administração e o Zabbix Agent coleta dados diretamente do host monitorado. Durante a fase de configuração inicial do ambiente, realizou-se o cadastramento do host principal que compõe a base da infraestrutura monitorada. Esse procedimento envolveu a definição do endereço IP, o grupo de monitoramento, os templates associados e demais parâmetros fundamentais para que o Zabbix Server pudesse coletar métricas de forma contínua. A interface correspondente a esse processo é apresentada na Figura 1, que demonstra visualmente a estrutura de configuração utilizada neste trabalho. Após a definição do host, procedeu-se à configuração dos itens responsáveis pela coleta das métricas analisadas ao longo do projeto. Os itens incluem parâmetros como ICMPPingLoss, ICMPPing, utilização de CPU, memória, disponibilidade de serviços e processos ativos. A Figura 2 ilustra essa etapa, exibindo a lista de itens habilitados. Nesse estágio, o sistema já era capaz de monitorar métricas como utilização de CPU, memória, processos ativos, latência e disponibilidade de serviços essenciais, utilizando tanto agentes nativos quanto protocolos complementares, como SNMP.

Além dos gatilhos padrões fornecidos pelo sistema, foi implementado um gatilho adicional para o monitoramento específico de perda de pacotes. Esse gatilho personalizado foi configurado para ser ativado automaticamente quando o Zabbix detecta perda de pacotes maior ou igual a 20% na máquina monitorada, permitindo identificar situações críticas de instabilidade de rede. A Figura 3 apresenta a interface de configuração dessas *triggers*, demonstrando as expressões lógicas usadas pelo sistema para identificar eventos críticos e ativar notificações.

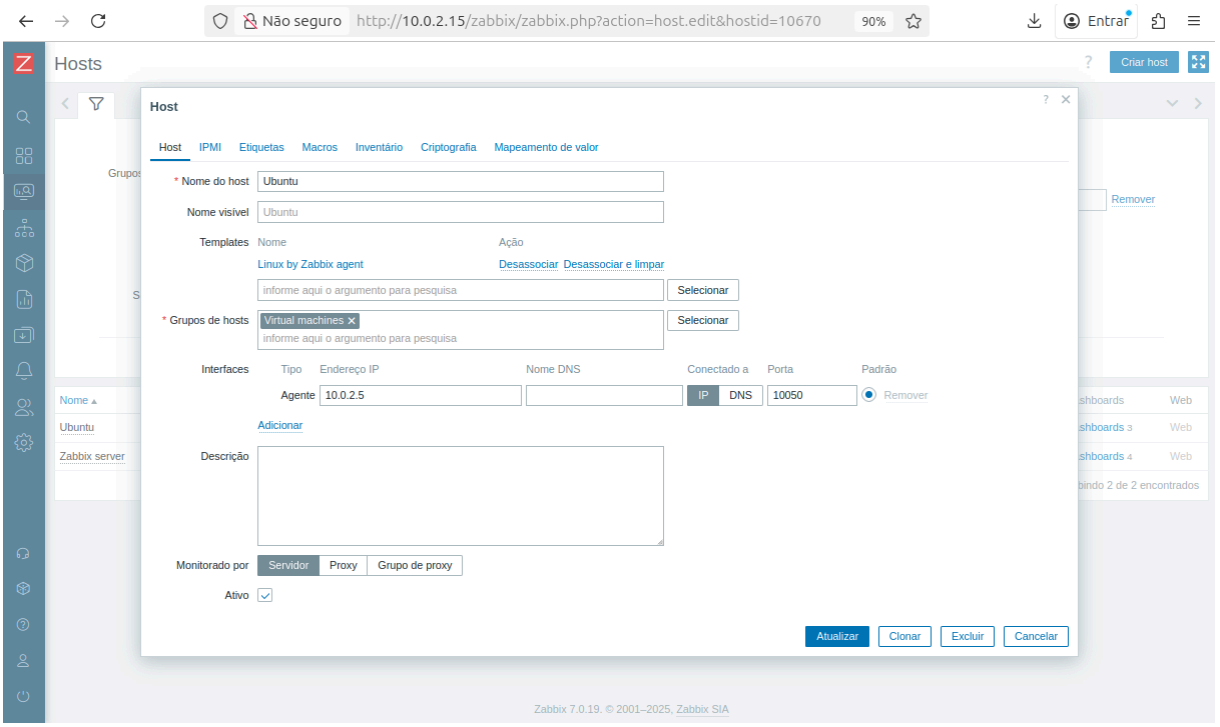
Para expandir o ambiente de simulação e validar o funcionamento do monitoramento distribuído, foi criada uma segunda máquina virtual, também baseada em Ubuntu Desktop 24.04 e configurada com os mesmos parâmetros técnicos da primeira máquina, porém com uma diferença fundamental: nesta segunda máquina foi instalado apenas o Zabbix Agent, sem os componentes de servidor e frontend. A segunda máquina passou a ser monitorada integralmente pelo Zabbix Server instalado na máquina principal, permitindo a coleta de métricas remotas, a execução de verificações padronizadas e a ativação de gatilhos de indisponibilidade.

Adicionalmente, implementou-se no Zabbix Server o recurso de envio automático de alertas por e-mail, utilizando o serviço SMTP do Gmail. Para isso, foi configurado um media type de e-mail com autenticação TLS, porta 587 e credenciais específicas, conforme exigido pelas políticas de segurança do provedor. Esse mecanismo permite que alertas gerados por gatilhos sejam encaminhados automaticamente à equipe responsável, ampliando a capacidade de resposta e estabelecendo um canal complementar às notificações via Telegram previstas nas etapas seguintes do desenvolvimento.

Com todos esses elementos, o Zabbix encontra-se operacional e funcionando de modo estável no ambiente de testes. O servidor central monitora simultaneamente a máquina local e a segunda máquina dedicada ao agente, fornecendo métricas contínuas ao Grafana e compondo a base para a automação de notificações que caracteriza a proposta deste trabalho.

A Figura 1 apresenta a interface de configuração do host monitorado no Zabbix, com os principais parâmetros necessários para sua inclusão no sistema.

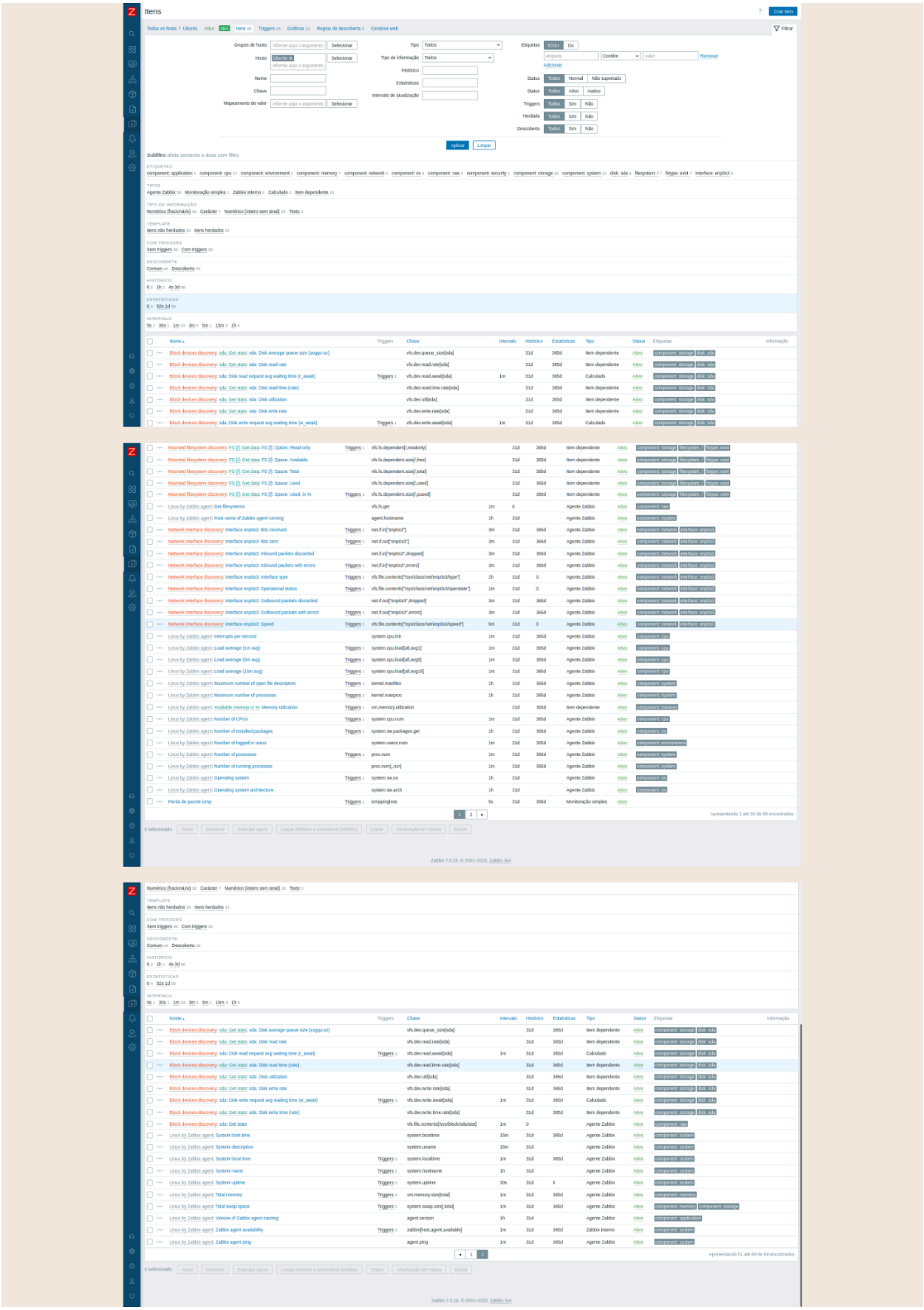
Figura 1 – Configuração do host monitorado no Zabbix



Fonte: Captura de tela do *software* Zabbix 7.0 LTS (Elaboração própria, 2025).

A Figura 2 ilustra os itens configurados no Zabbix responsáveis pela coleta das métricas monitoradas no sistema.

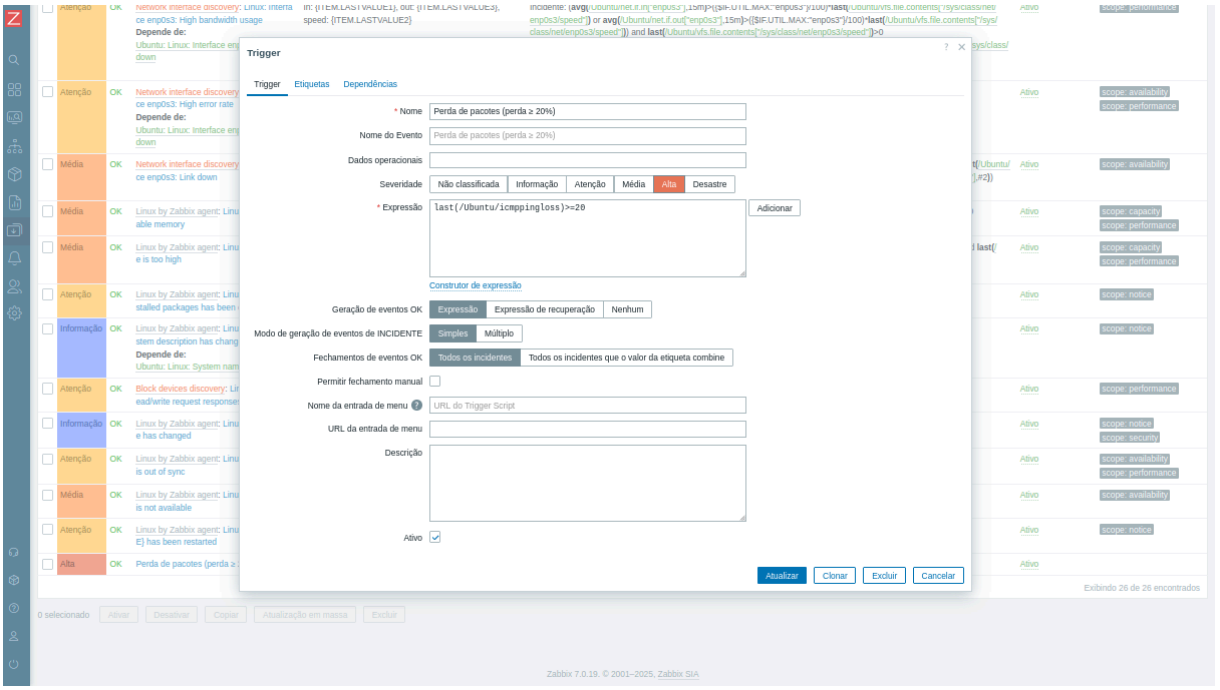
Figura 2 – Estrutura de itens configurados para coleta de métricas no Zabbix



Fonte: Capturas de tela do *software* Zabbix 7.0 LTS (Elaboração própria, 2025).

A Figura 3 demonstra a configuração dos gatilhos utilizados para a detecção automática de falhas e anomalias.

Figura 3 – Configuração de *triggers* responsáveis pela detecção de falhas



Fonte: Captura de tela do *software* Zabbix 7.0 LTS (Elaboração própria, 2025).

2.2.2 Grafana

O Grafana é uma plataforma de visualização e análise de dados amplamente utilizada em ambientes de monitoramento e observabilidade. Desenvolvido como solução de código aberto, permite a construção de painéis interativos capazes de representar métricas em séries temporais de forma clara e dinâmica. Sua arquitetura modular possibilita a integração com diversas fontes de dados, incluindo bancos de dados relacionais, sistemas de monitoramento e APIs especializadas, o que o torna versátil em cenários complexos de infraestrutura. No contexto deste projeto, o Grafana desempenha o papel de camada visual responsável por traduzir métricas técnicas em representações gráficas acessíveis, facilitando a interpretação rápida de eventos, tendências e anomalias na rede monitorada.

Embora o Zabbix ofereça visualizações nativas, estas tendem a ser técnicas e pouco flexíveis. O Grafana foi incorporado ao projeto para suprir essa limitação, atuando como camada de visualização gráfica interativa. Segundo Souza, Hanna e

Acosta (2020), o Grafana destaca-se pela capacidade de criar *dashboards* personalizáveis baseados em séries temporais de dados, o que o torna ideal para o acompanhamento em tempo real de variáveis de desempenho da rede.

A instalação do Grafana foi realizada na mesma máquina virtual utilizada para hospedar o Zabbix Server, também executando o sistema operacional Ubuntu Desktop 24.04. A escolha por instalar ambos os sistemas na mesma máquina deveu-se à praticidade e à etapa inicial de validação do protótipo; entretanto, ressalta-se que o Grafana pode operar em um host independente, desde que inserido na mesma rede e com permissão de acesso ao servidor Zabbix. A instalação ocorreu mediante o uso do repositório oficial disponibilizado pela Grafana Labs, seguindo um procedimento padrão: adição do repositório, atualização dos pacotes do sistema e instalação do pacote principal do Grafana. A versão instalada do Grafana foi a 12.2.0. Após a instalação, o serviço foi ativado e configurado para inicialização automática, mantendo-se todas as definições padrão de porta (3000/TCP), diretórios internos e permissões de execução.

Uma vez operacional, o Grafana foi integrado ao Zabbix para permitir o consumo direto das métricas coletadas. O dashboard empregado neste trabalho não foi desenvolvido integralmente do zero, tratando-se de uma adaptação de um painel pré-existente disponibilizado pela comunidade Grafana, intitulado *Zabbix – Full Server Status*, desenvolvido por Paulo Paim e posteriormente atualizado para compatibilidade com o Zabbix 7 e o Grafana 11. O modelo original foi modificado para atender às necessidades específicas do sistema proposto, incluindo ajustes na organização dos painéis, seleção de métricas e adequação dos limiares de monitoramento (Paim, 2018).

A construção do *dashboard* de monitoramento, apresentada na Figura 4, sintetiza esse objetivo ao reunir diferentes painéis que exibem, em séries temporais, dados como utilização de CPU e memória, perda de pacotes e disponibilidade de hosts. Essa integração foi realizada por meio da instalação do plugin oficial “Zabbix Data Source”, que funciona como uma API intermediária entre as duas plataformas. Após a instalação do plugin via interface administrativa do Grafana, foi configurada uma nova fonte de dados apontando diretamente para o Zabbix Server, mediante uso das credenciais de acesso ao frontend e da URL interna da instância Zabbix. Esse processo, ilustrado na Figura 5, consistiu em estabelecer a comunicação direta entre o Grafana e o Zabbix Server. O plugin utiliza a API nativa do Zabbix para

realizar consultas, possibilitando que o Grafana obtenha séries temporais, valores históricos, medições em tempo real e estados de *triggers*.

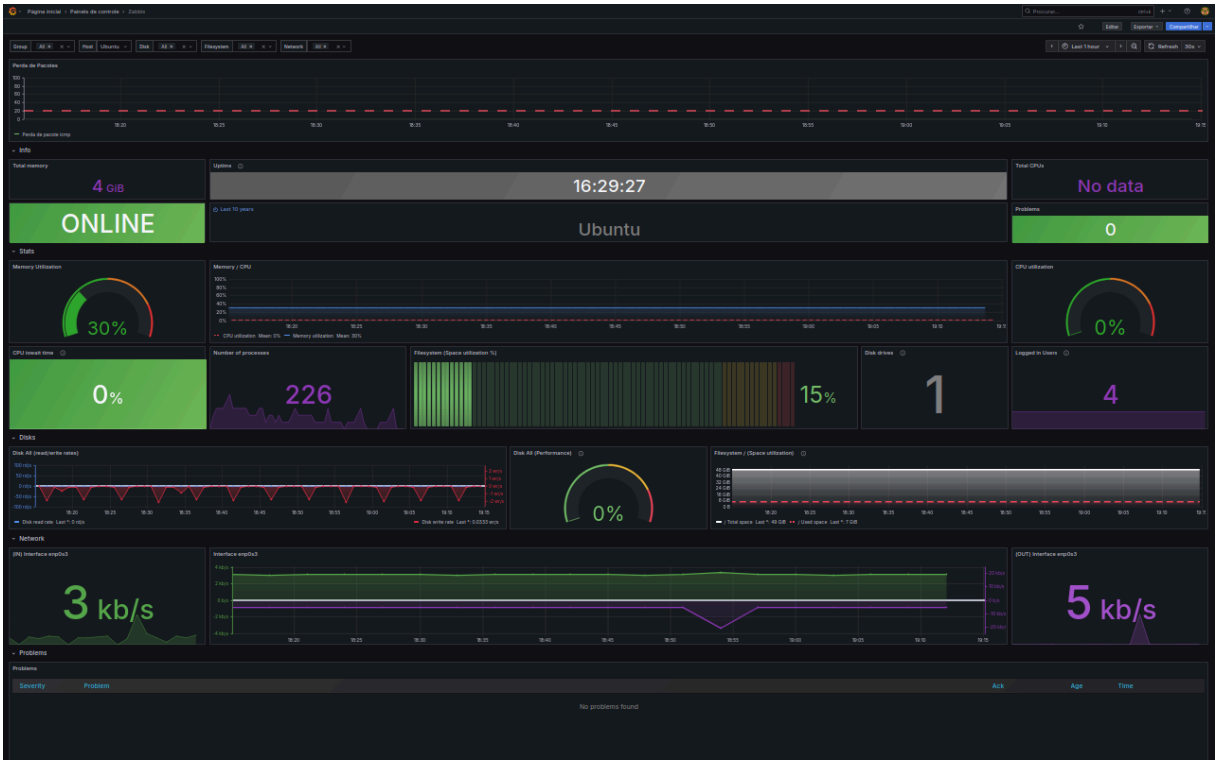
Essa integração habilita o Grafana a exibir, de forma dinâmica e visualmente intuitiva, métricas como disponibilidade de hosts, utilização de CPU e memória, latência, perda de pacotes e status de serviços monitorados. Durante as etapas de teste, foram construídos painéis experimentais que reúnem essas informações, demonstrando a capacidade da ferramenta de traduzir dados técnicos em representações visuais claras, contribuintes para uma análise mais rápida e eficiente por parte da equipe técnica da instituição.

Com base nesses gatilhos foi concluída também a integração entre Grafana, Zabbix e Telegram, possibilitando o envio de alertas automatizados sempre que o sistema identifica anomalias relacionadas aos eventos monitorados. O Telegram passou a atuar como canal de notificação rápida, garantindo que os administradores recebam avisos instantâneos, mesmo fora do ambiente de trabalho, favorecendo ações proativas de manutenção. A Figura 6 apresenta a configuração interna dos alertas no Grafana.

Com o Grafana instalado, integrado ao Zabbix e conectado ao sistema de alertas via Telegram, consolidou-se a camada de visualização e notificação do sistema de monitoramento. Essa estrutura permite acompanhar métricas em tempo real de forma organizada e intuitiva, contribuindo para a detecção precoce de falhas e para a tomada de decisões técnicas fundamentadas.

A Figura 4 apresenta o dashboard de monitoramento desenvolvido no Grafana para visualização das métricas coletadas.

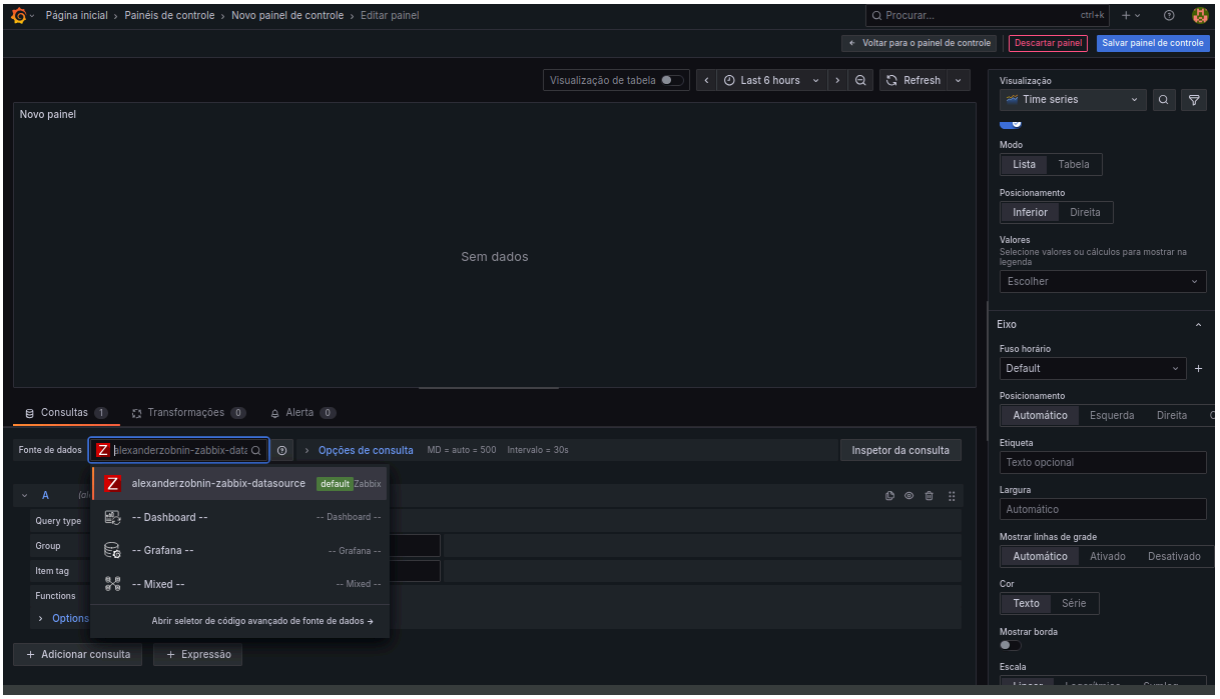
Figura 4 – *Dashboard de monitoramento elaborado no Grafana (adaptado de Paim, 2018)*



Fonte: Captura de tela do *software* Grafana 12.2.0 (Elaboração própria, 2025).

A Figura 5 ilustra a integração do Zabbix ao Grafana como fonte de dados para os painéis de monitoramento.

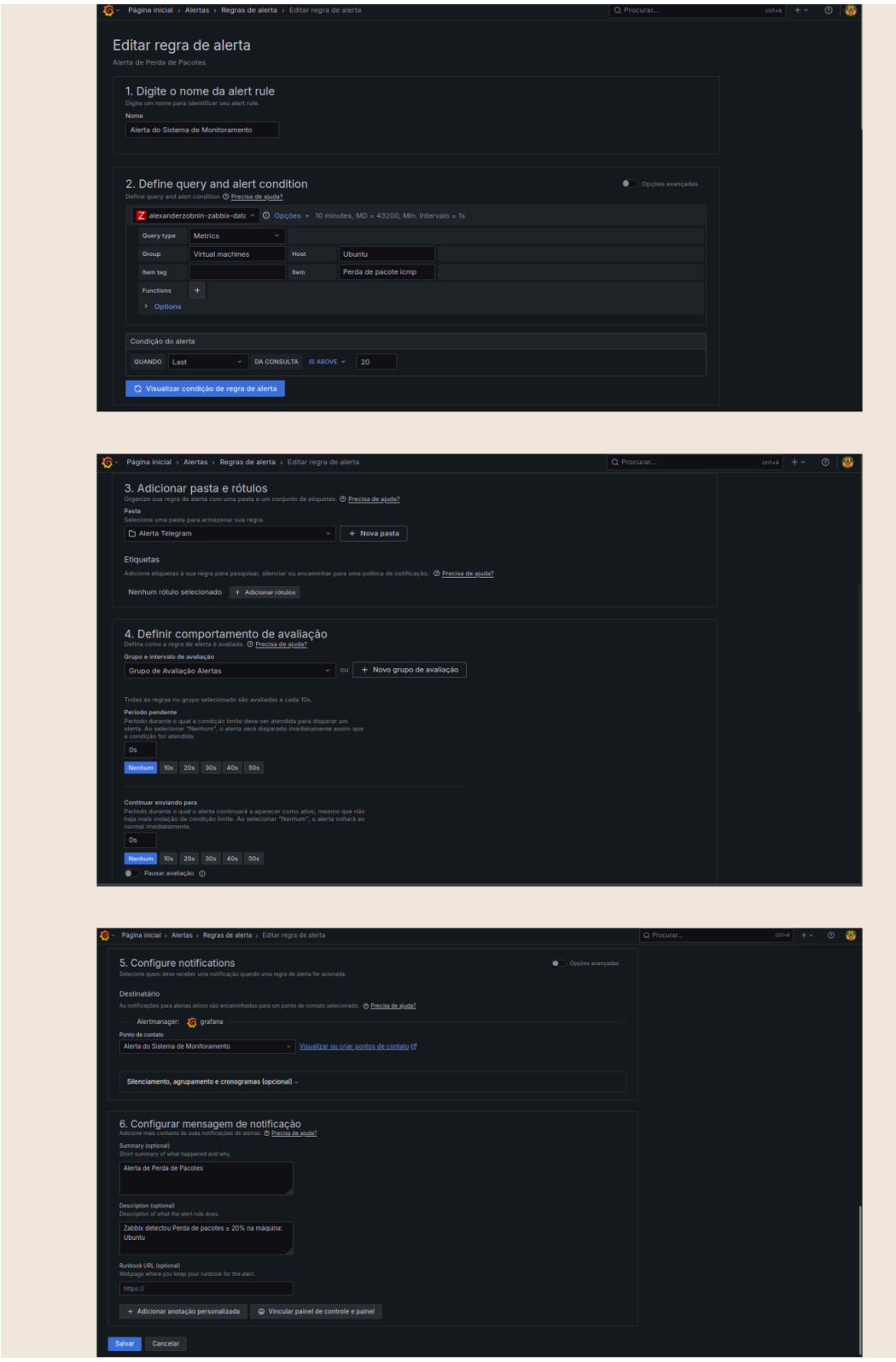
Figura 5 – Integração do Zabbix como fonte de dados no Grafana



Fonte: Captura de tela do *software* Grafana 12.2.0 (Elaboração própria, 2025).

A Figura 6 apresenta a configuração dos alertas e do canal de notificação no Grafana.

Figura 6 – Configuração de alertas e canal de notificação no Grafana



Fonte: Capturas de tela do *software* Grafana 12.2.0 (Elaboração própria, 2025).

2.2.3 Telegram

O Telegram é uma plataforma de mensagens instantâneas baseada em nuvem, reconhecida por sua velocidade, segurança e capacidade de operar em múltiplos dispositivos de forma simultânea. Diferentemente de aplicativos tradicionais, ele oferece uma Interface de Programação de Aplicações (API) aberta que permite o desenvolvimento de *bots* automatizados capazes de interagir com usuários, grupos e sistemas externos. Essa característica o torna especialmente adequado para aplicações de monitoramento, já que possibilita o envio de notificações técnicas em tempo real, sem custos adicionais e com alta confiabilidade.

No contexto deste trabalho, o Telegram foi adotado como ferramenta central de comunicação automática entre o sistema de monitoramento e a equipe técnica, permitindo que alertas fossem recebidos de forma imediata e organizada. Sua capacidade de operar por meio de bots personalizados e de permitir integração direta com sistemas como o Grafana tornou-o um componente essencial para garantir a eficiência e a agilidade na resposta a incidentes.

O Telegram foi integrado ao sistema como um canal de envio de alertas automatizados, desempenhando uma função estratégica na comunicação imediata entre o sistema de monitoramento e a equipe técnica responsável. Sua adoção baseou-se na disponibilidade de uma API robusta, gratuita e amplamente documentada, que permite a criação de bots capazes de interagir diretamente com ferramentas externas utilizando requisições HTTP autenticadas.

A integração foi realizada por meio do plugin nativo de notificações do Grafana, que oferece suporte direto ao envio de mensagens para o Telegram sem necessidade de *scripts* externos. Inicialmente, foi criado um *bot* dedicado utilizando o BotFather, ferramenta oficial da plataforma destinada ao gerenciamento e registro de *bots*, responsável por gerar um *token* exclusivo de autenticação. A Figura 7 ilustra esse processo inicial de registro do *bot*, evidenciando o procedimento realizado dentro do próprio Telegram. Em seguida, identificou-se o *chat* ID do grupo destinado a receber as notificações, permitindo que o *bot* atuasse como emissor autorizado dentro de um canal coletivo, garantindo que todos os administradores recebessem simultaneamente os alertas técnicos.

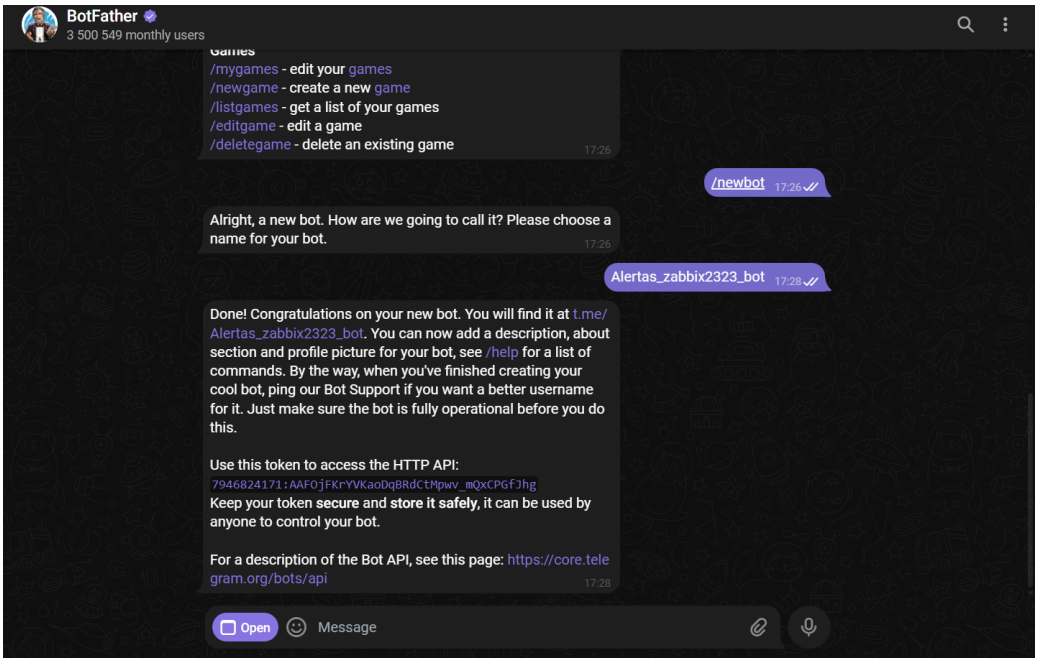
No Grafana, configurou-se um Canal de Notificação do tipo Telegram, acessível na seção de *Alerting*. A Figura 8 apresenta essa interface de configuração, demonstrando como o *bot* é incorporado ao sistema de notificações do Grafana. Nesse módulo, foram inseridos o token do *bot* e o *chat* ID do grupo, habilitando o canal como destino padrão para alertas provenientes dos *dashboards* monitorados. O *plugin* utiliza a API oficial do Telegram, estabelecendo comunicação com o *endpoint* "*sendMessage*" por meio de requisições HTTP POST, contendo informações como título do evento, descrição da anomalia, severidade, data e origem da métrica.

Uma vez finalizada a integração, o Grafana passou a encaminhar automaticamente mensagens para o grupo no Telegram sempre que qualquer regra de alerta configurada em *dashboards* fosse acionada. A Figura 9 apresenta um exemplo da mensagem de alerta recebida no Telegram, demonstrando o funcionamento da integração. Dessa forma, eventos como indisponibilidade de *hosts*, variações anormais de latência, sobrecarga de recursos ou perda de pacotes passaram a ser comunicados imediatamente, assegurando maior velocidade na detecção e resolução de incidentes.

A adoção do Telegram como meio de notificação aprimorou significativamente a responsividade do sistema. Por ser uma plataforma multiplataforma, leve e de alta disponibilidade, garantiu a entrega instantânea das mensagens, mesmo fora do ambiente institucional.

A Figura 7 demonstra o processo de criação do bot no Telegram por meio do BotFather.

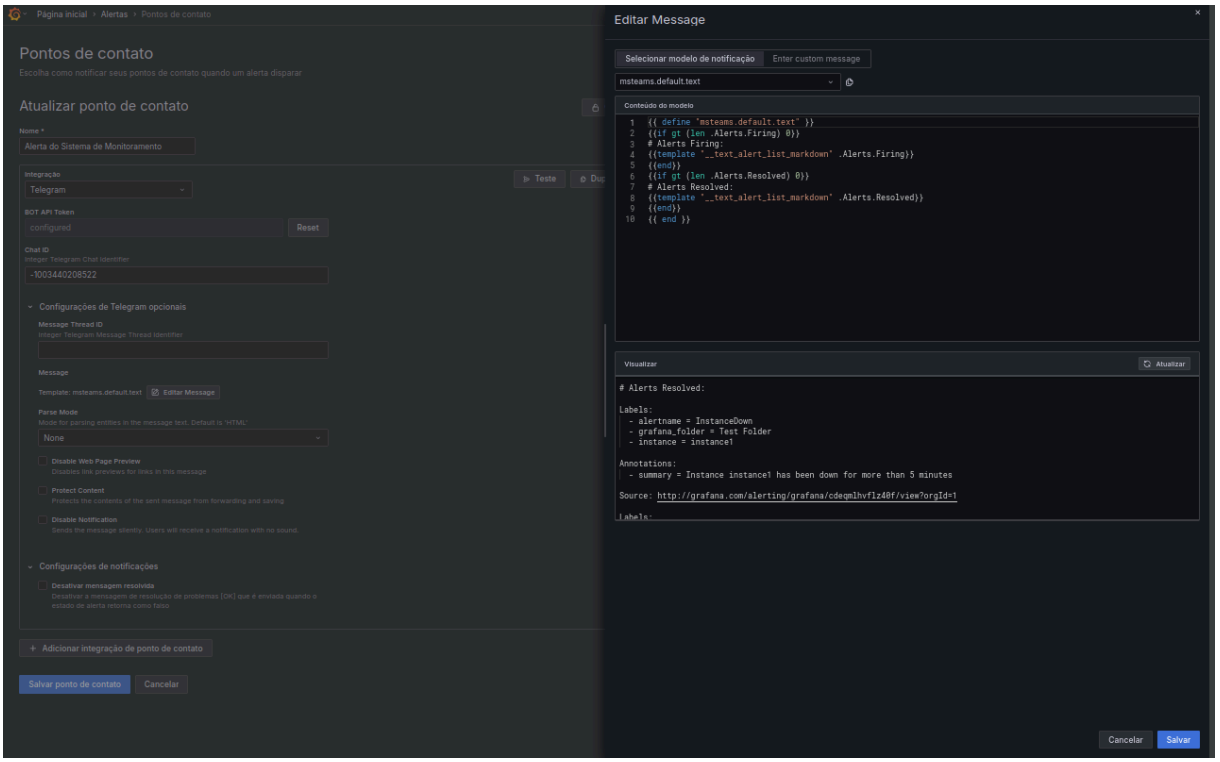
Figura 7 – Criação do bot no Telegram utilizando o BotFather



Fonte: Captura de tela do aplicativo Telegram (Elaboração própria, 2025).

A Figura 8 apresenta a configuração do bot do Telegram integrado ao Grafana.

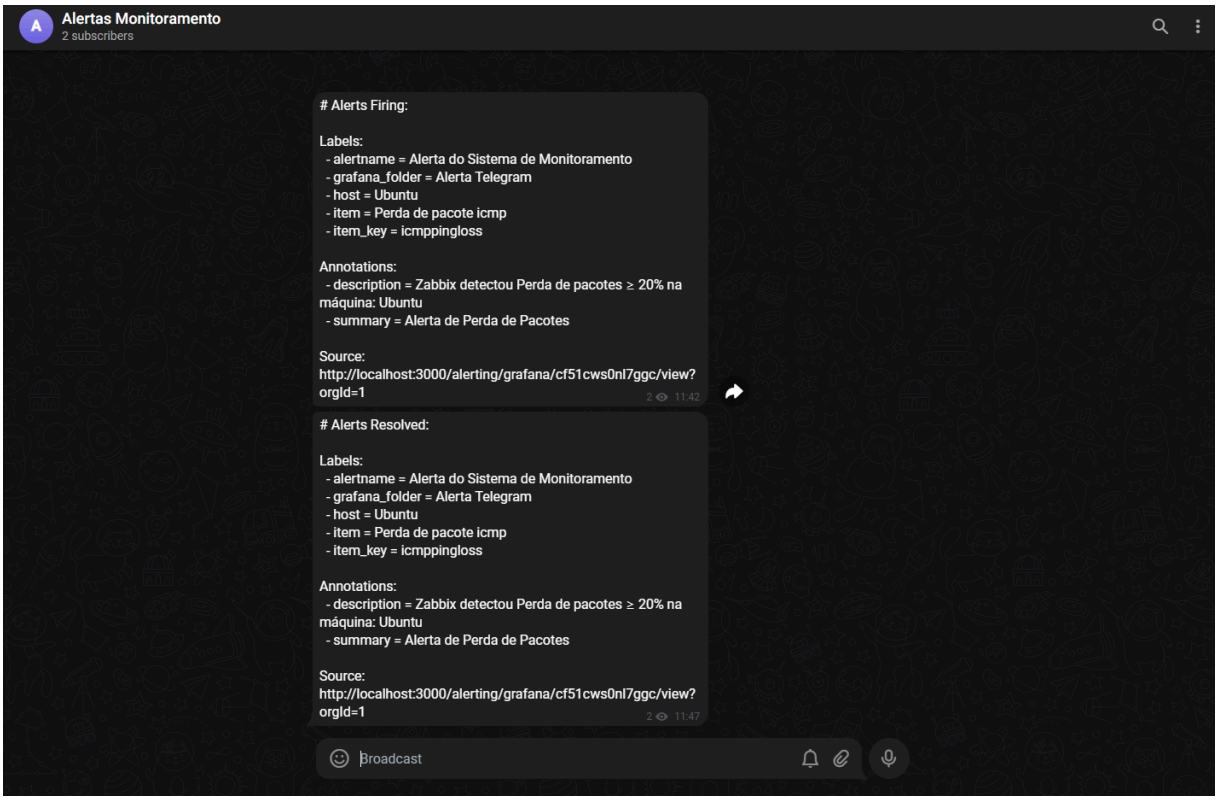
Figura 8 – Configuração do bot do Telegram com o Grafana



Fonte: Captura de tela do *software* Grafana 12.2.0 (Elaboração própria, 2025).

A Figura 9 ilustra um exemplo de alerta automatizado recebido no Telegram durante os testes do sistema.

Figura 9 – Exemplo de alerta automatizado recebido no Telegram



Fonte: Captura de tela do aplicativo Telegram (Elaboração própria, 2025).

2.2.4 pfSense

O pfSense é uma solução de firewall de código aberto baseada no sistema FreeBSD, amplamente utilizada para o gerenciamento e controle do tráfego de redes (Sena e Dutra, 2021). No contexto institucional do IFFar – Campus Uruguaiana, o pfSense constitui um elemento essencial da infraestrutura de TI, desempenhando funções como filtragem de pacotes, aplicação de regras de segurança, tradução de endereços (NAT), controle de VPNs e segmentação lógica da rede interna. Sua robustez, capacidade de customização e estabilidade tornam-no uma ferramenta consolidada para ambientes que demandam segurança e alta disponibilidade.

Apesar de sua relevância no ambiente físico da instituição, o pfSense não foi implementado neste projeto. A decisão fundamentou-se em dois fatores principais. Primeiramente, não havia necessidade técnica de incluir o *firewall* no ambiente de

testes, uma vez que os objetivos desta fase concentravam-se na validação do fluxo de monitoramento entre o Zabbix, o Grafana e os hosts monitorados. Como o foco era demonstrar a coleta de métricas, a integração entre as ferramentas e o funcionamento dos alertas automáticos, a presença do *firewall* não alteraria os resultados, já que suas funções não interferem na comunicação interna do Zabbix em ambientes controlados.

Em segundo lugar, a inclusão do pfSense exigiria a criação de uma terceira máquina virtual dedicada, dado que o *firewall* necessita de acesso direto a pelo menos duas interfaces de rede virtuais para operar adequadamente. Entretanto, a estação física utilizada para executar a simulação não dispunha de capacidade de processamento suficiente para hospedar simultaneamente três máquinas virtuais em operação contínua com níveis adequados de desempenho. A adição do pfSense implicaria risco de degradação do ambiente e poderia comprometer a coleta de métricas durante os testes.

Importante destacar que essa decisão não compromete a validade nem a aplicabilidade do sistema proposto. Caso a solução venha a ser instalada em ambiente de produção, o pfSense poderá coexistir plenamente com o sistema de monitoramento sem qualquer tipo de conflito. Isso ocorre porque o Zabbix utiliza protocolos e portas amplamente compatíveis com *firewalls* corporativos, tais como o tráfego HTTP/HTTPS do Frontend, o acesso à API e as portas de comunicação do Zabbix Agent. O pfSense, por atuar como camada de segurança de borda, não interfere na comunicação interna entre hosts autenticados dentro da rede, desde que as regras de *firewall* permitam o tráfego necessário.

2.3 ATIVIDADES REALIZADAS

Dando continuidade ao plano metodológico previamente definido, foram executadas diversas atividades técnicas que fundamentam e estruturam o desenvolvimento do sistema de monitoramento proposto. As ações realizadas contemplam desde o estudo teórico das ferramentas até a implementação e integração prática dos componentes no ambiente virtualizado.

Inicialmente, procedeu-se à revisão bibliográfica, com foco em trabalhos voltados ao uso de ferramentas livres de monitoramento, incluindo Zabbix, Grafana, pfSense e mecanismos de notificação automatizada via Telegram. Esse

levantamento permitiu identificar abordagens já adotadas em ambientes institucionais, compreender limitações de propostas anteriores e consolidar boas práticas técnicas relevantes ao desenvolvimento deste sistema. Os estudos de Souza, Hanna e Acosta (2020), Santos (2011), Andreoli (2016) e Sena e Dutra (2021) contribuíram significativamente para a construção da base conceitual, destacando o potencial de integrações entre Zabbix, Grafana e ferramentas de alerta para ambientes educacionais.

Na etapa seguinte, foi realizada a análise técnica das ferramentas selecionadas, avaliando-se compatibilidade, requisitos operacionais, desempenho e documentação disponível. Foram estudados os métodos de integração entre o Zabbix e o Grafana por meio do plugin oficial Zabbix Data Source, bem como as possibilidades de envio automatizado de alertas via Telegram utilizando a API nativa da plataforma. Em paralelo, avaliou-se o papel do pfSense no ambiente institucional real, embora ele não tenha sido incluído no ambiente de testes devido à ausência de necessidade funcional e limitações de hardware para sua virtualização simultânea.

A terceira fase consistiu na configuração do ambiente virtualizado utilizado para os testes. Com o auxílio do Oracle VirtualBox, foram implementadas duas máquinas virtuais baseadas em Ubuntu Desktop 24.04.2 (64 bits). A primeira máquina foi utilizada para hospedar o Zabbix Server, o Zabbix Frontend, o Zabbix Agent, o banco de dados MySQL e o servidor Apache, além da instalação do Grafana. A segunda máquina foi configurada unicamente com o Zabbix Agent, servindo como host monitorado remoto. Esse arranjo possibilitou a simulação de um ambiente distribuído e funcional, permitindo validar a comunicação entre agentes e servidor, o processamento de métricas e a operação dos gatilhos.

A etapa subsequente envolveu a instalação e configuração completa dos componentes centrais. O Zabbix 7.0 LTS foi implantado seguindo as diretrizes oficiais da Zabbix SIA, permitindo o monitoramento de métricas como uso de CPU, memória, latência, disponibilidade e processos ativos. Além dos gatilhos padrão, foi criado um gatilho personalizado para detecção de perda de pacotes igual ou superior a 20%, contribuindo para avaliar cenários de instabilidade de rede dentro da simulação. Também foi implementado um media type para envio de alertas via e-mail utilizando SMTP com autenticação TLS do Gmail, estabelecendo um segundo canal de notificação.

Em paralelo, procedeu-se à instalação do Grafana e à configuração da integração com o Zabbix por meio do plugin oficial Zabbix Data Source. Esse conector habilitou o acesso direto às métricas armazenadas no servidor, permitindo a construção de *dashboards* experimentais com visualização em tempo real de disponibilidade, latência, uso de CPU e memória.

Concluída a integração entre Zabbix e Grafana, iniciou-se a implementação da camada de notificações automáticas via Telegram. Primeiramente, foi criado um bot utilizando o BotFather, obtendo-se o token de autenticação necessário. Em seguida, identificou-se o chat ID do grupo destinado a receber os alertas, permitindo que o bot atuasse como remetente autorizado em um canal coletivo. No Grafana, foi configurado um Notification Channel do tipo Telegram, inserindo-se o token e o chat ID do grupo. A partir dessa configuração, o Grafana passou a enviar, automaticamente, mensagens para o Telegram sempre que regras de alerta configuradas fossem acionadas. Esse mecanismo utiliza a API oficial da plataforma, conectando-se ao endpoint sendMessage via requisições HTTP POST.

Atualmente, o ambiente de testes encontra-se plenamente funcional. O fluxo de dados entre Zabbix e Grafana opera de forma estável, os *dashboards* exibem métricas em tempo real e o sistema de alertas via Telegram já se encontra configurado e apto a enviar mensagens automáticas para o grupo técnico. A integração entre os módulos demonstra viabilidade técnica e consistência para futura implantação no ambiente de produção do Instituto Federal Farroupilha – Campus Uruguaiana.

2.3.1 Testes e Validação

Com o ambiente devidamente configurado e integrado, procedeu-se à realização de testes controlados com o objetivo de validar o funcionamento do sistema de monitoramento, dos gatilhos configurados e do mecanismo de envio de alertas automáticos. Os testes foram conduzidos de forma intencional e planejada, simulando cenários comuns de falhas em redes de computadores, a fim de verificar a capacidade do sistema em detectar anomalias e notificar corretamente a equipe técnica.

O primeiro teste consistiu na simulação de perda de pacotes na máquina monitorada. Para isso, foram utilizados comandos nativos do sistema operacional

Linux, aplicados diretamente no host monitorado, com o intuito de gerar artificialmente perdas de pacotes ICMP. Inicialmente, foi configurada uma perda de pacotes de aproximadamente 40%, valor superior ao limiar definido no gatilho personalizado do Zabbix, que estava configurado para disparar alertas quando a perda atingisse ou ultrapassasse 20%. Nessa condição, o sistema apresentou comportamento conforme o esperado: o Zabbix identificou a anomalia, o gatilho foi acionado corretamente, os dados foram refletidos nos dashboards do Grafana e os alertas automáticos foram enviados ao grupo configurado no Telegram, comprovando o funcionamento integrado de todas as camadas do sistema.

Na sequência, realizou-se um segundo cenário de teste de perda de pacotes, desta vez configurando uma degradação de aproximadamente 10%. Esse valor foi propositalmente mantido abaixo do limite estabelecido no gatilho, com o objetivo de validar se o sistema evitaria falsos positivos. Conforme esperado, o Zabbix registrou a métrica de perda de pacotes, porém o gatilho não foi acionado e nenhum alerta foi enviado. Esse comportamento confirmou que a lógica de detecção estava corretamente ajustada e que o sistema respondia apenas a situações consideradas críticas segundo os parâmetros definidos.

Durante a execução desses testes, algumas dificuldades foram identificadas. Destaca-se a necessidade de ajustes finos nos comandos de simulação de perda de pacotes no Linux, uma vez que pequenas variações nos parâmetros podiam resultar em oscilações nos valores medidos pelo Zabbix. Além disso, observou-se que o intervalo de coleta das métricas influenciava diretamente o tempo de disparo dos alertas, exigindo a calibração adequada dos tempos de atualização para garantir maior precisão e confiabilidade nos testes.

O segundo teste realizado teve como objetivo simular a queda total de conectividade com a internet. Nesse cenário, a interface de rede da máquina monitorada foi desativada manualmente, interrompendo completamente a comunicação com o servidor Zabbix. Como resultado, o gatilho configurado para detecção de indisponibilidade de host foi acionado, indicando corretamente a queda de conectividade. Assim como no teste anterior, o evento foi refletido nos dashboards do Grafana e os alertas automáticos foram enviados com sucesso via Telegram, notificando a ocorrência da falha de forma imediata.

Os resultados obtidos a partir desses testes demonstram que o sistema desenvolvido é capaz de identificar tanto degradações parciais quanto falhas totais de conectividade, acionando alertas de maneira consistente e confiável.

3 CONSIDERAÇÕES FINAIS

O presente Trabalho de Conclusão de Curso teve como objetivo propor e desenvolver, em ambiente virtualizado, um sistema de monitoramento de redes voltado ao Instituto Federal Farroupilha – Campus Uruguaiana, fundamentado na integração de ferramentas de código aberto, como Zabbix, Grafana e Telegram. Essa integração visou proporcionar uma abordagem automatizada e visualmente acessível para a identificação de falhas, o acompanhamento de métricas e o envio de alertas em tempo real à equipe técnica responsável pela infraestrutura de rede da instituição.

Durante o processo de desenvolvimento, foi possível validar a viabilidade técnica da integração entre as ferramentas e constatar a estabilidade do fluxo de dados entre o Zabbix e o Grafana. O ambiente virtualizado configurado no Oracle VirtualBox demonstrou-se adequado para simulações controladas, permitindo a realização de testes sem riscos à rede institucional. Ademais, o uso de tecnologias livres reafirmou-se como alternativa eficiente e economicamente viável, especialmente em contextos públicos.

Entretanto, alguns desafios influenciaram diretamente o andamento do projeto. A escassez de referências específicas sobre integrações práticas entre Zabbix, Grafana e Telegram dificultou a construção de uma base teórica sólida, exigindo maior esforço na investigação de documentações técnicas dispersas e em fóruns comunitários. Soma-se a isso a dificuldade inicial de manipulação de sistemas baseados em Linux, cuja lógica de comandos e estrutura de permissões demandou um período maior de adaptação e aprofundamento para que as configurações fossem realizadas de maneira consistente e segura. Esses obstáculos, apesar de impactarem o desenvolvimento, contribuíram para o aprimoramento do conhecimento técnico e para a consolidação das soluções apresentadas.

Os resultados evidenciam que a proposta é tecnicamente exequível e que sua futura implantação em ambiente de produção poderá contribuir significativamente para a otimização da gestão de redes no campus, reduzindo o tempo de resposta a incidentes e aumentando a confiabilidade da infraestrutura tecnológica. O sistema proposto representa, portanto, um avanço no sentido de uma gestão proativa e descentralizada, promovendo maior eficiência operacional e autonomia institucional.

Como trabalhos futuros, sugere-se o desenvolvimento de relatórios automatizados de desempenho e disponibilidade, gerados periodicamente e encaminhados aos administradores responsáveis, de modo a facilitar auditorias e análises históricas da rede. Em uma etapa mais avançada, considera-se viável a expansão da solução para múltiplos campi do Instituto Federal Farroupilha, permitindo o monitoramento centralizado em um único painel administrativo integrado.

Por fim, espera-se que este trabalho sirva como base para aprimoramentos contínuos e inspire novos projetos na área de administração de redes, fortalecendo a cultura de inovação, autonomia tecnológica e adoção de soluções abertas no âmbito institucional.

REFERÊNCIAS

ANDIFES – Associação Nacional dos Dirigentes das Instituições Federais de Ensino Superior. Disponível em: https://www.andifes.org.br/. Acesso em: jul. 2025.

ANDREOLI, Yuri. Análise comparativa entre ferramentas para gerenciamento e monitoramento de redes. 2016. 105 f. Trabalho de Conclusão de Curso (Graduação em Tecnologia em Análise e Desenvolvimento de Sistemas) – Universidade Tecnológica Federal do Paraná, Câmpus Pato Branco, 2016. Disponível em: http://repositorio.utfpr.edu.br/jspui/handle/1/15568. Acesso em: maio 2025.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. ABNT NBR 10520: Informação e documentação – Citações em documentos – Apresentação. Rio de Janeiro: ABNT, 2002.

ASSOCIAÇÃO NACIONAL DOS DIRIGENTES DAS INSTITUIÇÕES FEDERAIS DE ENSINO SUPERIOR (ANDIFES). *Nota da Andifes sobre a situação do orçamento das universidades federais*. Brasília, 14 maio 2025. Disponível em: https://www.andifes.org.br/2025/05/14/nota-da-andifes-sobre-situacao-do-orcamento-das-universidades-federais/. Acesso em: jul. 2025.

GRAFANA. Grafana documentation. [S. l.], [s. d.]. Disponível em: https://grafana.com/docs/grafana/latest/fundamentals/. Acesso em: maio 2025.

NETGATE. pfSense documentation. [S. l.], [s. d.]. Disponível em: https://docs.netgate.com/pfsense/en/latest/. Acesso em: maio 2025.

PAIM, Paulo. Zabbix – Full Server Status. [S. l.], 2018. Dashboard eletrônico. Disponível em: https://grafana.com/grafana/dashboards/5363-zabbix-full-server-status/. Acesso em: maio 2025.

PANDORA FMS. Flexible Monitoring System. [S. l.], [s. d.]. Disponível em: https://pandorafms.com/. Acesso em: jul 2025.

SANTOS, Raquel Ferraz Cunha. Gerência e monitoramento de redes. 2011. Trabalho de Conclusão de Curso (Tecnologia em Segurança da Informação) – Faculdade de Tecnologia de Americana, Americana, 2011. Disponível em: http://ric.cps.sp.gov.br/handle/123456789/1481. Acesso em: maio 2025.

SENA, João Vitor Souza; DUTRA, Wenedy Douglas. Gerenciamento e monitoramento de redes usando Zabbix e Grafana. 2021. Trabalho de Conclusão de Curso (Sistemas de Informação) – Faculdade Doctum, [S. l.], 2021. Disponível em: https://dspace.doctum.edu.br/handle/123456789/4810. Acesso em: maio 2025.

SOUZA, Davi Machado de; HANNA, Maicon Warthmann; ACOSTA, Roberto Bartzen. Gerenciamento e monitoramento de redes com Zabbix. Revista Acadêmica Alcides Maya, v. 2, n. 2, 24 jul. 2020. Disponível em:

https://raam.alcidesmaya.edu.br/index.php/RAAM/article/view/214. Acesso em: maio 2025.

TELEGRAM. Telegram Bot API Documentation. [S. l.], [s. d.]. Disponível em: https://core.telegram.org/bots/api. Acesso em: out 2025.

TELEGRAM. Telegram FAQ – Bots. [S. l.], [s. d.]. Disponível em: https://core.telegram.org/bots. Acesso em: out 2025.

ZABBIX. Zabbix documentation 7.0. [S. l.], [s. d.]. Disponível em: https://www.zabbix.com/documentation/7.0/pt/manual. Acesso em: maio 2025.

ZDNET. Network downtime: costs and impacts. 2023. Disponível em: https://www.zdnet.com/article/network-downtime-costs-and-impacts/. Acesso em: jul 2025.