

Arquitetura e Métodos para Monitoramento e Análise de Desempenho em Redes Open RAN: Uma Implementação no Testbed OpenRAN@Brasil

Janssen Martins^{1,2}, Murilo Silva¹, Elenice Pedrosa¹, Ronikleyton Lima¹, Luan Rios¹, Gabriel Vassoler¹, Joaquim Ribeiro¹, Cesar Gama¹, Lucas Borges^{1,4}, Diego Abreu¹, Gustavo Araújo^{1,3}, Lucas Bondan^{1,3}, Roger Immich², Fernando Farias¹

¹Rede Nacional de Ensino e Pesquisa

²Universidade Federal do Rio Grande do Norte (UFRN)

³Universidade de Brasília (UnB)

⁴Universidade Federal do Pará (UFPA)

{janssen.martins,murilo.silva,elenice.pedrosa,lucas.oliveira}@rnp.br
{ronikleyton.lima,gabriel.vassoler,joaquim.ribeiro,diego.abreu}@rnp.br
{cesar.ferreira,luan.rios,gustavo.araujo,lucas.bondan}@rnp.br
roger@imd.ufrn.br, fernando.farias@rnp.br

Abstract. *The advent of Open RAN networks has introduced a disaggregated and programmable architecture, yet it has brought significant challenges for monitoring large-scale distributed infrastructures. This work presents an observability solution implemented in the RNP's OpenRAN@Brasil Testbed, integrating IT infrastructure health with granular indicators from the srsRAN software stack. The proposed architecture utilizes a topology based on Zabbix Proxies to ensure telemetry constraints, a specialized monitoring daemon was developed to work alongside the Zabbix Sender, performing the programmatic extraction of indicators from the RU, DU, and CU units. Experimental results demonstrate that the solution is non-intrusive as Zabbix process consumes an average of 4.90% CPU and 53.1 MiB of RAM, while the customized daemon operates with an average of 2.1% CPU and only 28.4 MiB of RAM. The solution allows for the continuous collection of power consumption, traffic, and antenna hardware usage metrics with WAN traffic lower than 130 kb/s.*

Resumo. *O advento das redes Open RAN introduziu uma arquitetura desagregada e programável, mas trouxe desafios significativos para o monitoramento de infraestruturas distribuídas em larga escala. Este trabalho apresenta uma solução de observabilidade implementada no ambiente de testes OpenRAN@Brasil do RNP, integrando a saúde da infraestrutura de TI com indicadores granulares da pilha de software srsRAN. A arquitetura proposta utiliza uma topologia baseada em proxies Zabbix para garantir restrições de telemetria. Um daemon de monitoramento especializado foi desenvolvido para funcionar em conjunto com o Zabbix Sender, realizando a extração programática de indicadores das unidades RU, DU e CU. Os resultados experimentais demonstram que a solução não é intrusiva, pois o processo Zabbix consome em média 4,90% da CPU e 53,1 MiB de RAM, enquanto o daemon personalizado opera com uma média de 2,1% da CPU e apenas 28,4 MiB de RAM. A solução permite a coleta contínua de métricas de consumo de energia, tráfego e uso de hardware de antena com tráfego WAN inferior a 130 kb/s.*

1. Introdução

A evolução das redes móveis de quinta geração (5G) consolidou a transição para arquiteturas abertas e desagregadas, impulsionada pela iniciativa Open RAN, que promove a interoperabilidade entre múltiplos fornecedores e a virtualização de funções de rede. Nesse ecossistema, a observabilidade torna-se um pilar crítico para a gestão operacional, exigindo mecanismos que transcendam o monitoramento convencional de TI para alcançar uma visão holística do continuum entre nuvem e borda [Mekki 2024].

Embora avanços recentes tenham focado na padronização de interfaces de controle, como o monitoramento unificado de modelos de serviço E2 [Silva et al. 2025], a implementação prática em larga escala ainda enfrenta desafios significativos. A literatura aponta que a sensibilidade de *stacks* de software aberto, como a srsRAN, às variações de recursos de hardware exige uma correlação precisa entre indicadores de desempenho de rádio e métricas de infraestrutura física, especialmente em cenários onde a distribuição geográfica dos nós da rede introduz latências e instabilidades que podem comprometer a integridade da telemetria.

Diante da complexidade de gerenciar essas infraestruturas distribuídas, surge a necessidade de soluções que integrem a gerência de falhas e desempenho de forma resiliente. Embora existam *daemons* voltados para a detecção de anomalias e segurança em redes abertas através da telemetria da *stack* srsRAN [Jere et al. 2024], há uma carência de implementações que validem a persistência desses dados em ambientes de larga escala com alta latência de gerência. Este trabalho propõe preencher essa lacuna ao apresentar uma arquitetura de monitoramento unificada, implementada sobre o Testbed Open-RAN@Brasil, que utiliza instâncias estrategicamente posicionadas de Zabbix Proxies para assegurar a coleta de métricas granulares.

Desta forma, se propõe uma abordagem sistemática para o monitoramento automatizado e a observabilidade de infraestruturas Open RAN distribuídas. Tendo foco na coleta granular de métricas da *stack* srsRAN em ambientes de pesquisa, e utilizando *scripts* personalizados, esta proposta permite a visibilidade em tempo real das funções de rede desagregadas. A solução oferece não apenas a extração de KPIs de rádio em tempo real, mas também a sua correlação direta com o consumo de recursos físicos dos servidores, fornecendo uma base de dados sólida para a orquestração e otimização da rede em cenários de nuvem distribuída. Os experimentos e a implementação visam identificar padrões de comportamento e gargalos operacionais, fornecendo diretrizes práticas para a manutenção e otimização de *testbeds* Open RAN de larga escala. Assim, os eixos principais desta contribuição são:

- Infraestrutura de Monitoramento Distribuída: Implementação de uma topologia baseada em Zabbix Proxies para as ilhas do Testbed OpenRAN@Brasil, permitindo a escalabilidade da coleta e a resiliência dos dados em um cenário de rede geograficamente dispersa.
- Automação da Telemetria RAN: Desenvolvimento de um *daemon* baseado em Python e Zabbix Sender para a extração programática de indicadores de desempenho (KPIs) das unidades RU, DU e CU, superando as limitações de protocolos de gerência convencionais em *stacks* de software aberto.
- Visibilidade e Análise de Desempenho: Consolidação de *dashboards* analíticos no Grafana que correlacionam o estado de saúde do hardware e do sistema operacio-

nal com métricas específicas de rádio, fornecendo uma ferramenta de diagnóstico para pesquisadores e operadores da infraestrutura.

A partir desse cenário, o presente trabalho busca responder a questões fundamentais sobre a observabilidade em redes abertas, investigando, primeiramente, como a correlação em tempo real entre o estado da infraestrutura física e os indicadores de desempenho da pilha de rádio pode auxiliar na identificação de gargalos operacionais em ambientes Open RAN. Adicionalmente, examinam-se as estratégias necessárias utilizadas para viabilizar a coleta automatizada de métricas granulares em componentes desagregados da RAN sem comprometer o desempenho das funções de rede. Por fim, aborda-se como garantir a integridade e a persistência de dados telemétricos em larga escala diante das latências e instabilidades inerentes a uma infraestrutura de rede distribuída geograficamente, como a apresentada pelo *testbed* OpenRAN@Brasil.

O presente artigo está organizado conforme segue. A Seção 2 detalha os trabalhos relacionados e as lacunas preenchidas por esta pesquisa. Em seguida, a Seção 3 apresenta a metodologia e o desenvolvimento do *daemon*. Os resultados obtidos no Testbed OpenRAN@Brasil são apresentados e discutidos na Seção 4. Por fim, a Seção 5 conclui o trabalho com as considerações finais e perspectivas de trabalhos futuros.

2. Trabalhos Relacionados

A análise de redes acadêmicas e de infraestruturas voltadas para tecnologias emergentes tem sido foco de diversos estudos. Embora existam propostas consolidadas sobre a utilização do Open RAN, existe uma lacuna referente a trabalhos que explorem como a observabilidade é essencial em redes 5G desagregadas e como integrar o monitoramento e a avaliação de tráfego em arquiteturas abertas introduz novos desafios. Isso se reflete de forma especial no que diz respeito a precisão das medições em tempo real e ao impacto das configurações de virtualização sobre o desempenho da rede. Nesse contexto, [Guemdani et al. 2025] conduziu experimentos e coleta das métricas de desempenho utilizando um conjunto de ferramentas em diferentes camadas do sistema, em específico empregando a ferramenta *iperf3* para a geração de tráfego de rede e a aferição da taxa de transmissão.

Paralelamente, o monitoramento contínuo das estatísticas de rede e a análise da largura de banda em tempo real foram realizados através de utilitários nativos de sistemas baseados em Linux, nomeadamente *ifstat*, *bmon* e *nload*. Por fim, para a extração e medição do consumo dos recursos de hardware, utilização de CPU e as operações de entrada e saída (I/O) em disco, empregou-se a biblioteca *psutil* da linguagem Python. Neste trabalho serão apresentadas alternativas para facilitar a observabilidade em ambientes de rede de acesso via rádio (RAN) aberta, onde a correlação entre a saúde da infraestrutura física e os indicadores de desempenho das funções virtuais tornam-se um requisito crítico.

2.1. Arquitetura de Monitoramento para Redes 5G

No contexto de redes 5G, [Silva et al. 2025] apresenta um modelo de monitoramento que utiliza a interface E2 em arquiteturas O-RAN, e faz a integração entre Zabbix e Grafana para a descoberta automatizada de E2 Service Models (E2SMs). A solução proposta foca na visibilidade das capacidades expostas pelos nós da rede (gNBs) e no suporte

ao desenvolvimento de xApps, validando a eficácia do uso de *scripts* personalizados e agentes Zabbix para consolidar métricas de rádio e de infraestrutura computacional em ambientes baseados em OpenAirInterface (OAI).

Adicionalmente, [Barrachina-Muñoz et al. 2022] implementa componentes de monitoramento como Funções de Rede Contêinerizadas (CNFs) dentro de um *cluster* Kubernetes, permitindo a coleta integrada de métricas tanto da infraestrutura computacional quanto do domínio de rádio (RAN). A proposta visa facilitar o gerenciamento do ciclo de vida da rede 5G e permitir que diferentes *stakeholders*, como operadores da infraestrutura e proprietários de *network slices* tenham acesso em tempo real aos Indicadores Chave de Desempenho (KPIs) essenciais para validar Acordos de Nível de Serviço (SLAs). A arquitetura de monitoramento proposta utiliza coleta unificada com o Prometheus, extraíndo automaticamente métricas da infraestrutura do Kubernetes, tais como CPU e memória via *kube-prometheus*, além de métricas específicas de rádio como taxa de bits, através de um *script* Python customizado que se conecta à estação base via *websocket*. De modo distinto, no presente trabalho empregou-se uma abordagem descentralizada, onde o pré-processamento de métricas é realizado localmente, e posteriormente as métricas são enviadas a um servidor Zabbix central. Também aplicou-se a visualização integrada com o Grafana para que os administradores da infraestrutura possam analisar simultaneamente os eventos da infraestrutura computacional e os sinais da rede de rádio.

Embora [Barrachina-Muñoz et al. 2022] compartilhe similaridades com a presente solução na *stack* tecnológica de monitoramento utilizando um *cluster* Kubernetes multi-nós, a proposta do presente trabalho inova ao aplicar conceitos de observabilidade em uma **infraestrutura real, dedicada e geograficamente distribuída**. Enquanto trabalhos anteriores utilizam ambientes emulados para os usuários (como o Amarisoft AMARI UE Simbox), este estudo aborda os desafios práticos de integridade e persistência de dados telemétricos nas múltiplas ilhas do Testbed OpenRAN@Brasil, utilizando uma arquitetura baseada em Zabbix Proxies para mitigar os impactos de latência e instabilidade de rede inerentes a um cenário de produção nacional.

Além disso, uma das principais inovações reside no desenvolvimento de um **daemon de extração assíncrona** voltado especificamente para a *stack* srsRAN. Diferente da abordagem de [Silva et al. 2025], que se concentra na compatibilidade de E2SMs via FlexRIC, a presente solução provê um mecanismo de coleta não intrusivo que correlaciona indicadores de desempenho de rádio com o estado da infraestrutura física em tempo real. Este *daemon* preenche uma lacuna na observabilidade de funções de rede virtualizadas ao permitir a extração de KPIs críticos via interface *websocket*, sem a necessidade de modificar a lógica interna da pilha srsRAN ou depender exclusivamente da maturidade da interface E2.

Além disso, uma das principais inovações reside no desenvolvimento de um *daemon* de extração assíncrona voltado especificamente para a *stack* srsRAN. Diferente da abordagem de [Silva et al. 2025], que se concentra na compatibilidade de E2SMs via FlexRIC, a presente solução provê um mecanismo de coleta não intrusivo que correlaciona indicadores de desempenho de rádio com o estado da infraestrutura física em tempo real, preenchendo uma lacuna prática na observabilidade de funções de rede virtualizadas que não dependem exclusivamente da interface E2 para a exposição de KPIs críticos. Neste artigo detalha-se como a coleta de métricas da pilha srsRAN é feita através de uma

interface *websocket* disponível para pilhas O-RAN.

2.2. Caracterização de Desempenho da *Stack* srsRAN

A análise de desempenho da *stack* srsRAN em ambientes virtualizados tem sido objeto de investigação recente. No *testbed* OpenRAN@Brasil, é possível utilizar uma infraestrutura de redes 5G e a desagregação de funções de rede mediante o uso de padrões abertos, a exemplo da pilha srsRAN, bem como realizar a análise de desempenho de diversos componentes de uma rede 5G. [De Oliveira et al. 2025] descrevem como se empregou a plataforma Nephio para realizar a orquestração baseada em intenção, simplificando a implantação automatizada de funções de núcleo e de borda em múltiplos *clusters* Kubernetes utilizando o *testbed* OpenRAN@Brasil. Ao integrar ferramentas de código aberto como Open5GS e srsRAN, o estudo prova a viabilidade técnica de gerenciar redes complexas e distribuídas de forma eficiente, porém não abordam como a observabilidade contínua pode ser adotada em ambientes complexos, e a importância de monitorar métricas em infraestruturas descentralizadas.

A monitoração do sistema srsRAN pode ser observada no estudo de [Hassouna et al. 2025] e é realizada por meio da interface padrão E2, que conecta a estação base ao Near-RT RIC. Essa monitoração é efetivada pela implantação de dois aplicativos (xApps) no Near-RT RIC, baseados no projeto de código aberto Open AI Cellular (OAIC). Ao utilizarem o KPIMON xApp, o modelo de serviço E2SM-KPM extrai continuamente a telemetria da rede. Além disso um xApp customizado é utilizado para monitorar em tempo real as tendências da taxa de erro de bloco (BLER) e do Esquema de Modulação e Codificação (MCS). A visualização e o monitoramento são feitos de forma integrada por meio do xApp customizado que executa um processo programado em *python*.

A inovação desta proposta frente a esses trabalhos reside na criação de um ecossistema de observabilidade que é simultaneamente especializado em rádio, e resiliente a distâncias geográficas. Através do uso estratégico de Zabbix Proxies, a solução assegura que a correlação entre o esgotamento de recursos físicos e a degradação dos KPIs de rádio seja mantida mesmo em uma infraestrutura nacional distribuída, onde a instabilidade dos links de gerência inviabilizaria métodos de coleta convencionais. A abordagem proposta no presente trabalho não utiliza interface E2 para coletar métricas, adotando a extração direta de métricas com *scripts* em *python*. Desta forma, se permite a exposição dos dados imediatamente para coletores padrão da indústria de TI, tais como o Zabbix, e se evita a complexidade de implantar e orquestrar um controlador Near-RT RIC, e desenvolver aplicativos xApps específicos e exclusivamente para a coleta de dados. Além disso, o uso de funções de amostragem customizadas, como direto a API de equipamentos via protocolo WebSocket, permite extrair métricas de rádio desses equipamentos físicos. Essa abordagem viabiliza o monitoramento sem depender de atualizações para o padrão O-RAN e abertura da interface E2 para estações de rádio comerciais, e facilita a integração com equipamentos proprietários.

3. Metodologia

A metodologia fundamenta-se na estruturação de um sistema de monitoramento para redes abertas e distribuídas, dividida em três eixos: a infraestrutura do Testbed OpenRAN@Brasil, onde os testes foram validados; a arquitetura de monitoramento baseada

em Zabbix Proxies, agentes e SNMP; e a estratégia de extração de métricas da *stack srs-RAN* via *daemon* customizado para integração e análise de dados.

3.1. Arquitetura do Testbed OpenRAN@Brasil

O Testbed OpenRAN@Brasil constitui uma infraestrutura de âmbito nacional projetada para preencher a lacuna entre a academia, centros de pesquisa e a indústria, fornecendo acesso aberto a componentes de RAN desfragmentados. Ao integrar ambientes heterogêneos com diversas condições geográficas, climáticas e operacionais, o *testbed* oferece um ambiente de alta fidelidade para validação experimental. Esta arquitetura distribuída permite a execução de testes em larga escala e a verificação de modelos teóricos sob restrições do mundo real, alinhando-se aos requisitos de interoperabilidade e programabilidade do paradigma O-RAN.

A topologia é estruturada em nós regionais distribuídos, denominados ilhas: uma ilha da Rede Nacional de Ensino e Pesquisa (RNP) hospedada no Centro Brasileiro de Pesquisas Físicas (CBPF), no Rio de Janeiro; e outra ilha no Centro de Pesquisa e Desenvolvimento (CPQD), em Campinas. Estas ilhas são interconectadas através do *backbone* de alto desempenho da RNP, com coordenação hospedada no Rio *Science Data Center* (RSDC) e Rede Rio. Como ilustrado na Figura 1, este polo atua como o núcleo do *testbed*, provendo a orquestração global e o gerenciamento das funções de rede desfragmentadas. Essa infraestrutura é formada por servidores robustos utilizados para hospedar o Core 5G e a pilha CU/DU do Testbed OpenRAN@Brasil, *switchs* programáveis, CPE (*Customer Premises Equipment*) e OLT (*Optical Line Terminal*), reproduzindo um ambiente verossímil que garante o isolamento e a reprodutibilidade de experimentos em 5G com alta capacidade de banda.

A composição técnica desses nós utiliza processadores de propósito geral (GPP) e rádios definidos por software (SDR), como modelos USRP, para implementar os blocos funcionais O-RU e O-DU. A pilha de software baseia-se primordialmente em soluções de código aberto, como OpenAirInterface (OAI) e srsRAN, criando um ambiente flexível para experimentação nas camadas de rede. A camada de orquestração, gerenciada via Kubernetes, possibilita a escalabilidade e o isolamento das funções de rede entre os sites distribuídos. O núcleo da inteligência do *testbed* reside na integração do RAN Intelligent Controller em tempo real aproximado (Near-RT RIC) para o gerenciamento automatizado de recursos de rádio, conforme [De Oliveira et al. 2025].

Por fim, esta versatilidade arquitetural suporta diferentes cenários de pesquisa, incluindo o fatiamento dinâmico de rede (*network slicing*) e a mitigação de interferência baseada em IA. Ao expor interfaces padronizadas, o OpenRAN@Brasil provê uma plataforma robusta para a avaliação da próxima geração de sistemas de telecomunicações 5G/6G autônomos e inteligentes.

3.2. Arquitetura do Monitoramento

A arquitetura de monitoramento proposta neste trabalho foi concebida com o objetivo de fornecer visibilidade contínua e granular sobre o desempenho dos diferentes componentes que compõem o ecossistema *Open RAN* do Testbed OpenRAN@Brasil. Considerando a natureza distribuída e desagregada das redes *Open RAN*, compostas por elementos como RUs, DUs, CUs, RIC e infraestrutura de nuvem, a solução foi projetada seguindo

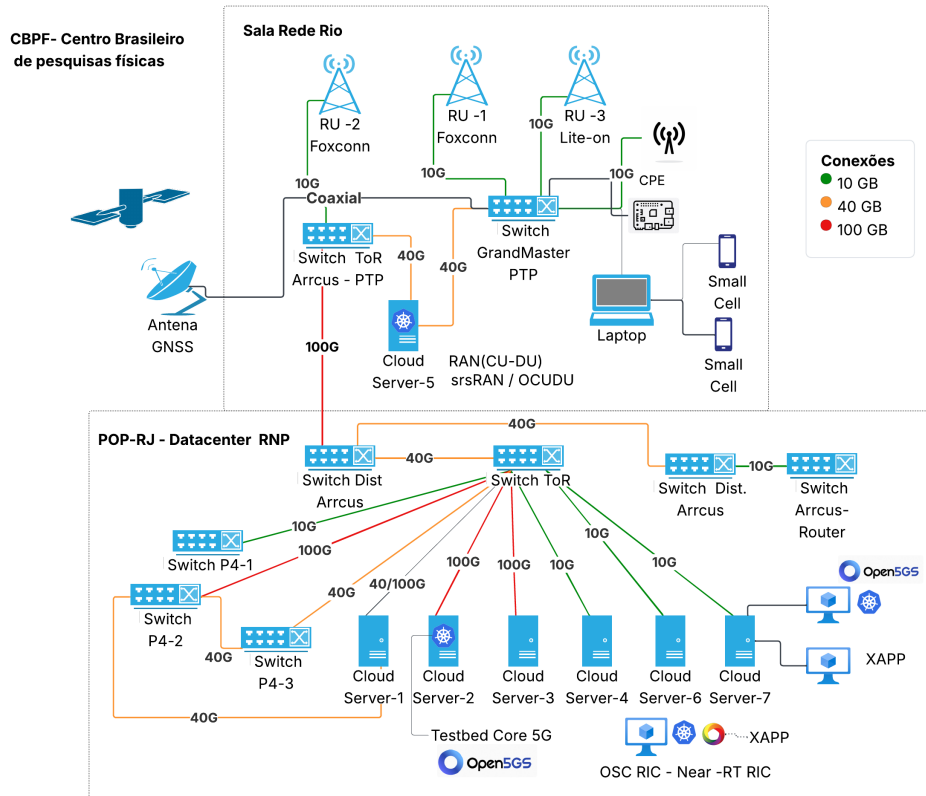


Figura 1. Arquitetura do Testbed OpenRAN@Brasil.

princípios de observabilidade, escalabilidade e baixo acoplamento entre os módulos de coleta, processamento e visualização de métricas.

Para suportar a dispersão geográfica e a heterogeneidade do ambiente, a solução utiliza uma arquitetura baseada em Zabbix e Grafana, estruturada com *proxies* em cada localidade remota. Essa abordagem descentraliza a coleta e processa métricas localmente, mitigando impactos de latência e garantindo a resiliência dos dados em cenários de instabilidade na conectividade. O monitoramento ocorre de forma híbrida: servidores e sistemas operacionais são integrados via Zabbix Agent 2 para extração de telemetria de baixo nível (CPU, memória e I/O), enquanto ativos de rede e hardware de suporte são monitorados via SNMP, permitindo uma visão consolidada da saúde dos dispositivos e do tráfego nas interfaces de comunicação.

No nível de agregação, os dados convergem para um servidor Zabbix central com banco de dados PostgreSQL, configurado para retenção de dados em 31 dias. A resiliência é garantida pelo armazenamento local nos *proxies* por até 24 horas em caso de falhas de conectividade, assegurando a integridade da telemetria. Conforme ilustrado na Figura 2, essa camada centraliza as informações dos domínios Open RAN distribuídos geograficamente entre as unidades na RNP (RJ), no CPQD (SP) e no PoP-SC da RNP (SC), provendo uma visão unificada do estado operacional do *testbed*.

A infraestrutura do servidor de monitoramento foi estabelecida para atuar como o ponto de convergência de toda a demanda de telemetria do Testbed OpenRAN@Brasil, incluindo a integração das novas ilhas de computação previstas para o projeto. Esta

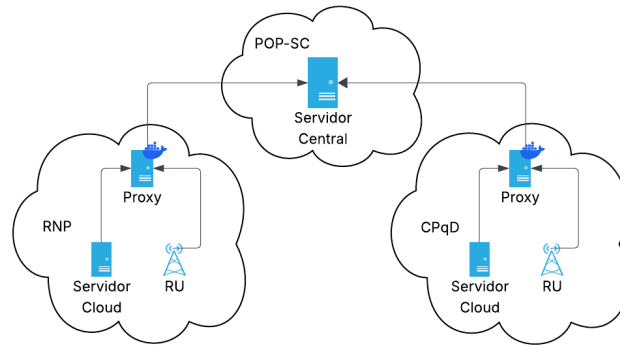


Figura 2. Arquitetura do Monitoramento.

configuração visa sustentar o volume de dados proveniente de múltiplas fontes heterogêneas, assegurando a estabilidade operacional e a confiabilidade das medições durante a execução de experimentos distribuídos. A Tabela 1 detalha as especificações técnicas deste servidor central, abrangendo os recursos de CPU, memória RAM, subsistema de armazenamento e o sistema operacional utilizado para suportar a carga de processamento da rede.

Tabela 1. Recursos do Servidor Central.

Componente	Descrição
Sistema Operacional	Debian 12 Bookworm
Memória RAM	DDR4-2666 16 GB
Modelo do processador	Intel Xeon Silver 4309Y 8 núcleos
Armazenamento	170GB HDD

3.3. Monitoramento das Funções de Rede O-RAN

O monitoramento das funções de rede O-CU (*Open Central Unit*) e O-DU (*Open Distributed Unit*) da arquitetura Open RAN, bem como dos planos de controle e usuário da O-RU (*Open Radio Unit*), foi realizado por meio de um *daemon*¹ desenvolvido e integrado às funções de rede Open RAN. Atualmente, o *daemon* suporta as funções de rede O-CU e O-DU do projeto srsRAN² v25.10, com suporte planejado às funções do projeto OCUDU³. Este *daemon* consiste em um *container* executado em paralelo ao processo das funções de rede, tendo como objetivo coletar, em tempo real, as métricas produzidas por essas funções, pré-processá-las e encaminhá-las ao servidor Zabbix.

Sua lógica de funcionamento é composta por três etapas principais, conforme ilustrado na Figura 3. Na primeira etapa, o *daemon* estabelece uma conexão WebSocket com a respectiva função de rede — O-CU ou O-DU — e, após o estabelecimento do canal de comunicação, envia uma solicitação de subscrição de métricas (`metrics_subscribe`). Essa subscrição é necessária para que a função de rede inicie o envio contínuo de métricas ao *daemon*. Na segunda etapa, as métricas recebidas são pré-processadas e adaptadas ao formato compatível com o Zabbix, operação realizada por meio da biblioteca

¹<https://git.rnp.br/openran/fase-1/testbed/sw/oran-zabbix-daemon>

²https://github.com/srsran/srsRAN_Project

³<https://gitlab.com/ocudu/ocudu>

zabbix-utils, disponibilizada oficialmente pela equipe de desenvolvimento do Zabbix. Por fim, na terceira etapa, os dados já processados são transmitidos ao servidor Zabbix para armazenamento e visualização.

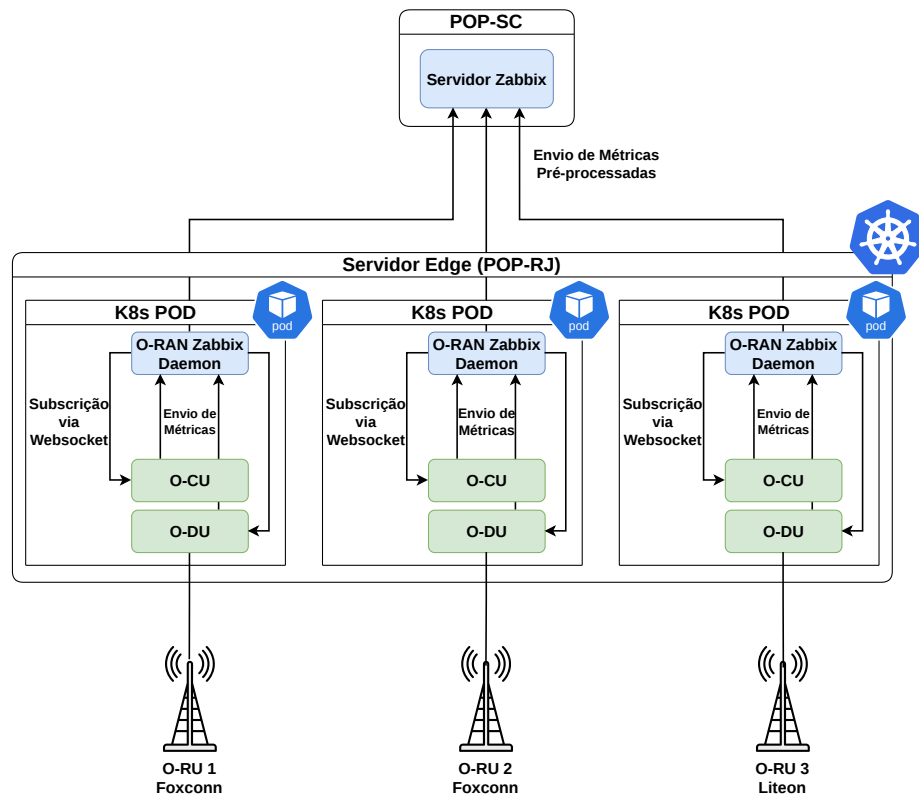


Figura 3. Arquitetura do Monitoramento das Funções de Rede O-CU e O-DU.

As métricas reportadas pelas funções de rede e monitoradas pelo O-RAN Zabbix Daemon podem ser organizadas em quatro grupos funcionais: `app_resource_usage`, `cells`, `du` (com ênfase em `du_high`) e `ru` (com ênfase em `ofh`). Essa estruturação permite observar, de forma integrada, indicadores de consumo computacional, qualidade de processamento por célula, desempenho por equipamento de usuário (UE) e comportamento temporal dos blocos de transporte fronthaul.

O grupo `app_resource_usage` concentra métricas globais do Servidor Edge, como utilização de CPU, consumo de memória e potência. O grupo `cells` organiza o monitoramento em dois níveis de granularidade: por célula, identificada pelo PCI (*Physical Cell Identity*), e por UE, identificado pelo RNTI (*Radio Network Temporary Identifier*). Nesse escopo, são coletados indicadores de latência média e máxima, falhas de alocação, volume de tráfego (*bitrate*), qualidade de canal e modulação (CQI, MCS), relação sinal-ruído (SNR) e contadores de sucesso e falha em transmissões nos sentidos downlink e uplink. Essa granularidade possibilita correlacionar degradações de desempenho com condições específicas de célula ou de usuário.

Para a O-DU, são capturadas métricas da subcamada `du_high.mac_dl`, com ênfase em latência de processamento — mínima, média e máxima — e utilização de

CPU por PCI. Para a O-RU, no domínio `ru.ofh`, são monitoradas latências e vazão dos blocos de transmissão e recepção Ethernet, tempos de codificação e decodificação de mensagens, bem como estatísticas de temporização e de pacotes — distinguindo, por exemplo, pacotes recebidos dentro do prazo (*on-time*), antecipados (*early*) e atrasados (*late*). Em conjunto, esse conjunto de métricas viabiliza análises de desempenho fim a fim, desde o processamento MAC até o transporte fronthaul.

A modelagem no Zabbix emprega itens do tipo *trapper* em combinação com o mecanismo de *Low-Level Discovery* (LLD), com regras de descoberta dinâmica parametrizadas por PCI e por par PCI/RNTI. Dessa forma, os itens de monitoramento são instanciados automaticamente de acordo com a topologia ativa e a carga observada em tempo de execução, eliminando configuração manual recorrente e conferindo maior escalabilidade à solução.

4. Resultados

A avaliação experimental da arquitetura proposta iniciou-se pela análise do impacto computacional da infraestrutura de monitoramento sobre os nós do Testbed OpenRAN@Brasil. Para quantificar o *overhead* introduzido, foram coletadas métricas de desempenho do *container* responsável pelo Zabbix Proxy durante um período de 4 horas de operação contínua em condições nominais de tráfego. Os dados, extraídos via *Docker stats*, revelaram um consumo médio de CPU de 5% e uma utilização de memória RAM de aproximadamente 1.6GiB. Estes resultados demonstram a eficiência da solução, indicando que a camada de persistência e intermediação de dados opera com baixo consumo de recursos, preservando a capacidade computacional dos servidores para a execução das pilhas de rádio virtualizadas e para o processamento de sinais em tempo real.

Adicionalmente, avaliou-se o impacto dos processos vinculados ao Zabbix Agent 2 através de uma configuração de auto-monitoramento, a qual verificou-se que o impacto operacional deste componente é mínimo (em condições de métricas realizadas de minuto em minuto). A Figura 4 apresenta os dados consolidados no *dashboard* do Grafana, fornecendo uma prova empírica da eficiência da solução. Os gráficos superiores detalham o consumo de recursos exclusivo do processo do Zabbix Agent 2, isolando o impacto da ferramenta de outros processos do sistema operacional.

Observa-se no gráfico superior que o uso de CPU pelo processo permanece estritamente linear, com uma média de 4,9%, enquanto o consumo de memória RAM (ao centro) apresenta oscilações controladas com uma média de 53MiB, confirmando o baixo impacto sobre o hardware. Complementarmente, o gráfico inferior monitora o tráfego na interface WAN, registrando médias de 40,5 kb/s para recebimento e 87,1 kb/s para envio. Esse volume reduzido de tráfego de gerência é essencial para a viabilidade do Testbed OpenRAN@Brasil, garantindo que a telemetria não onere os links de transporte de longa distância e nem concorra com o tráfego de dados da rede 5G. Estes índices reforçam a viabilidade da utilização de agentes de telemetria em ambientes de produção, uma vez que o *overhead* gerado é insuficiente para causar contenção de recursos ou interferir na estabilidade do processamento de rádio em tempo real, mesmo sob condições de alta granularidade de coleta.

Além do monitoramento específico dos processos de gerência, a arquitetura fornece uma visão abrangente da utilização de recursos de hardware em cada servidor de nu-

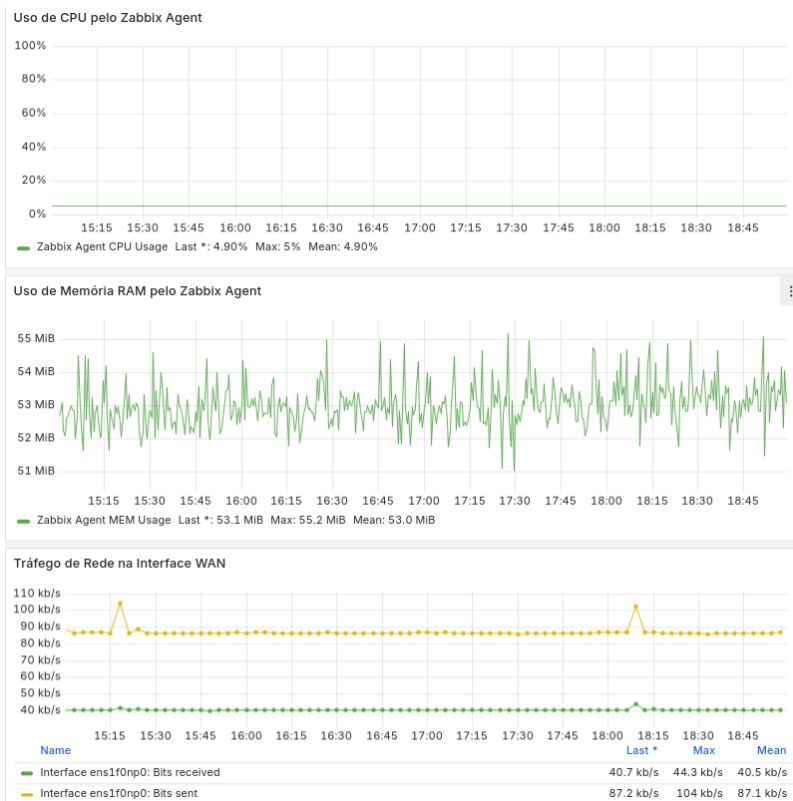


Figura 4. Monitoramento do Processo Zabbix Agent 2.

vem (*cloud node*). Como ilustrado na Figura 5, o *dashboard* consolida métricas críticas que representam o estado global do sistema, evidenciando uma disponibilidade de recursos confortável para a execução de experimentos. Observa-se que a utilização de memória RAM mantém-se em 26,3%, enquanto a ocupação de disco no diretório raiz está em 48,5%. O gráfico de CPU revela um uso médio de apenas 2,35%, com picos que não ultrapassam 4,32%, confirmando que o servidor possui ampla folga computacional. Esse monitoramento em tempo real, aliado ao registro de um *uptime* superior a 13 dias, é fundamental para garantir a estabilidade do Testbed OpenRAN@Brasil, permitindo que pesquisadores identifiquem preventivamente gargalos de infraestrutura que poderiam comprometer a precisão dos resultados dos experimentos.

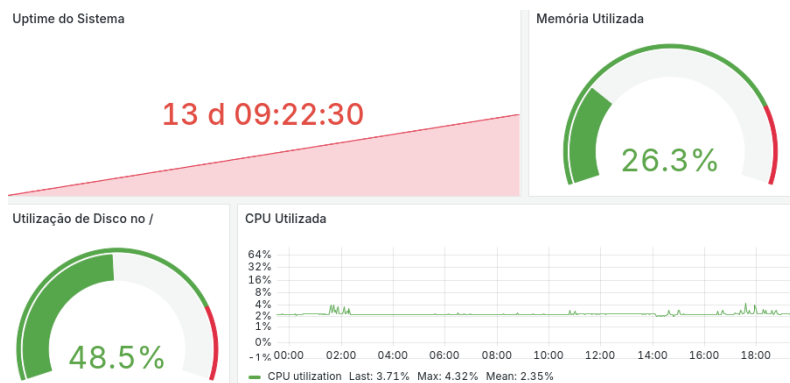


Figura 5. Monitoramento dos Recursos de Hardware dos Servidores de *Cloud*.

Consecutivamente, os dados consolidados na Figura 6 apresentam um cenário que indica a capacidade da arquitetura proposta em correlacionar a atividade de rádio com o consumo de infraestrutura no Testbed OpenRAN@Brasil. Os gráficos ilustram o comportamento do sistema durante um pico de utilização provocado por um Equipamento de Usuário (UE) operando em alta banda. Observa-se no gráfico superior que o tráfego de *downlink* (DL) salta de zero para um patamar próximo a 985 Mb/s no instante 13:15:30. Este evento desencadeia uma reação imediata no consumo de recursos pelas funções de rede (CU/DU): o uso médio de CPU eleva-se para 316%, valor que representa a utilização agregada de múltiplos núcleos de processamento, equivalente ao consumo integral de 3,16 CPUs, atingindo picos de até 551% (5,51 CPUs). Concomitantemente, o gráfico de consumo energético revela um salto de 217 W para uma média de 224 W, com picos de 233 W durante o estresse de tráfego. A estabilidade da memória RAM em 2,87 GiB durante todo o processo confirma que a oscilação de consumo é puramente transiente e vinculada ao processamento de pacotes, validando a precisão do *daemon* customizado na captura dessas variações em tempo real.

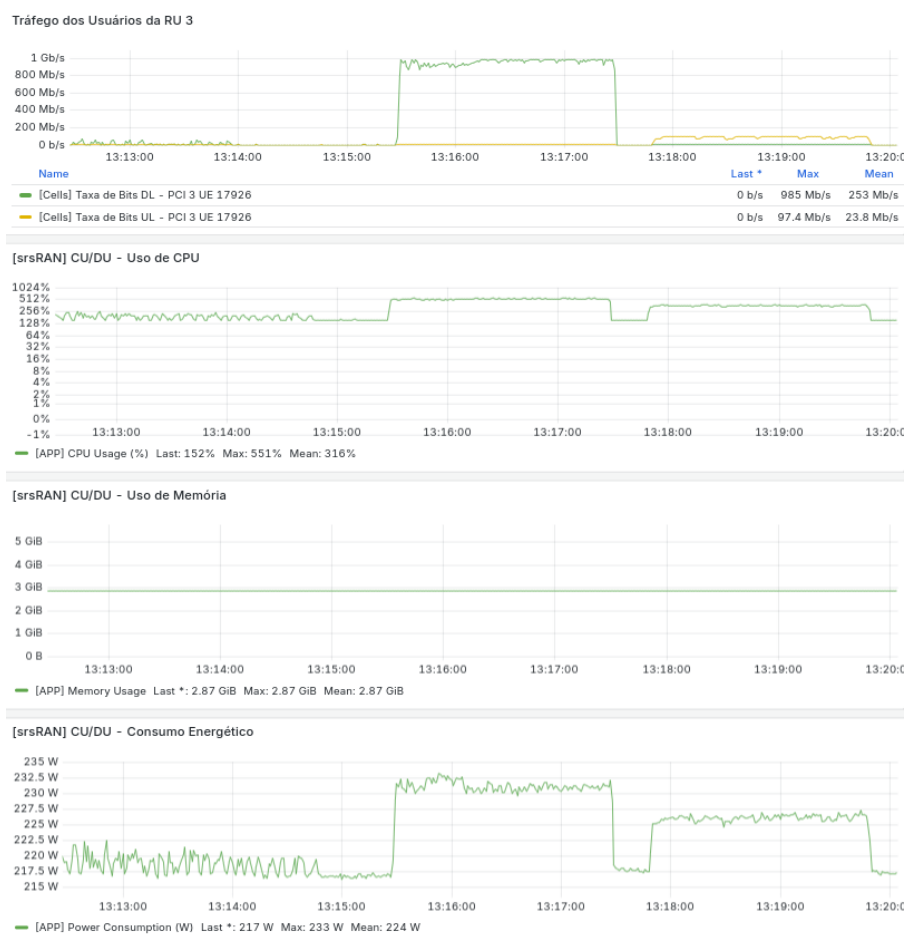


Figura 6. Monitoramento das Funções de Rede pelo *Daemon* Customizado.

Em última análise, a Figura 7 exibe a validação do impacto computacional da solução proposta, apresentando o consumo de recursos exclusivo do processo do *daemon* de monitoramento especializado. Durante o período de operação normal sob análise, observa-se no gráfico superior um comportamento estável, com consumo de memória

RAM oscilando de forma controlada entre 28 MiB e 30 MiB, registrando uma média de 28,4 MiB. Concomitantemente, o gráfico inferior detalha que o uso de CPU pelo processo permanece praticamente linear, com uma média de 2,10% e picos que não excedem 3,10%. Estes índices de consumo, extremamente reduzidos, corroboram a viabilidade da implementação do *daemon* no ambiente de produção do Testbed OpenRAN@Brasil. A alocação mínima de recursos indica que a telemetria não denigre a performance das funções de rede virtualizadas, preservando a integridade do processamento de rádio em tempo real e evitando a contenção de hardware que poderia comprometer os experimentos de terceiros na infraestrutura da RNP.

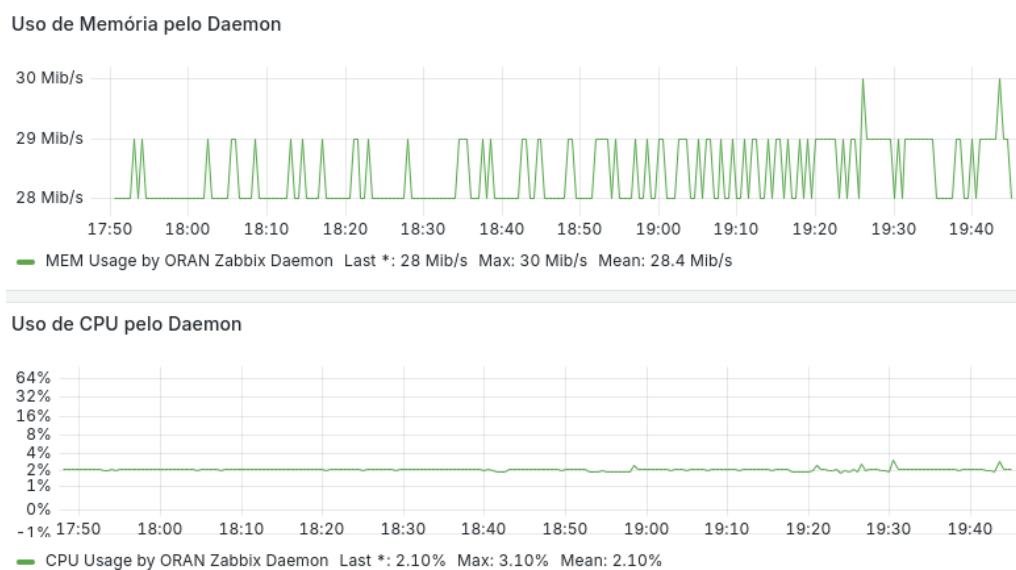


Figura 7. Consumo de Recursos de Hardware pelo *Daemon* Customizado.

Em síntese, a convergência das métricas de infraestrutura, telemetria de rádio e consumo energético em um plano de visualização único sugere a eficácia da arquitetura baseada em Zabbix e no *daemon* customizado. A capacidade de observar, com granularidade, que um fluxo de dados próximo a 1 Gbps demanda o processamento equivalente a 5,51 CPUs e eleva o consumo de hardware em aproximadamente 16 W (comparado ao estado inicial), fornece dados valiosos para futuras estratégias de automação.

5. Conclusão e Trabalhos Futuros

Este trabalho consolidou uma contribuição prática e reproduzível para a observabilidade de ambientes Open RAN ao propor uma arquitetura de monitoramento integrada, que abrange desde a infraestrutura física até as funções virtualizadas da rede. A adoção do Zabbix como plataforma unificada, potencializada pelo desenvolvimento do *O-RAN Zabbix Daemon*, permitiu centralizar métricas heterogêneas com granularidade por célula e por dispositivo de usuário (UE). Os resultados experimentais confirmaram a alta eficiência da solução, visto que o impacto computacional do *daemon* mostrou-se desprezível, apresentando uma média de 2,1% de uso de CPU e apenas 28 MiB de memória RAM. Esses índices validam uma abordagem de telemetria não intrusiva que preserva a integridade do processamento de rádio em tempo real, mesmo sob condições de estresse com tráfego de dados próximo a 1 Gbps.

Como desdobramentos futuros, pretende-se evoluir a arquitetura para suportar a federação das novas ilhas de computação e rádio que serão integradas ao Testbed OpenRAN@Brasil, utilizando a topologia de Zabbix Proxies para assegurar a resiliência da telemetria e a persistência dos dados em cenários de alta latência. Paralelamente, pretende-se disponibilizar um *template* público oficial para o monitoramento da pilha srs-RAN, incluindo arquivos de configuração, mapas de métricas e *dashboards* analíticos pré-configurados, visando fomentar a reprodutibilidade e a adoção da solução por outros pesquisadores da área.

Ademais, duas frentes de evolução estão previstas para o projeto *O-RAN Zabbix Daemon*. A primeira consiste na extensão do suporte às funções de rede O-CU e O-DU do projeto OCUDU, o que permitirá ao *daemon* operar de forma interoperável em relação à implementação das funções de rede, ampliando sua aplicabilidade em diferentes cenários de implantação. A segunda frente consiste na expansão do *parser* de métricas para abranger as camadas ainda não mapeadas, preservando o paradigma já adotado de descoberta dinâmica e padronização de chaves. Em conjunto, essas evoluções tendem a elevar a cobertura observável do sistema e a viabilizar análises fim a fim mais robustas, incluindo correlações entre os planos de controle, usuário e processamento de rádio, complementando, assim, o monitoramento já estabelecido da infraestrutura física e fortalecendo a capacidade de operação do ambiente do Testbed OpenRAN@Brasil.

Referências

- Barrachina-Muñoz, S., Manges-Bafalluy, J., and Payaró, M. (2022). Cloud-native 5g experimental platform with over-the-air transmissions and end-to-end monitoring. (4):1–6.
- De Oliveira, L. B., Da Silva, M. C., Marques, D. D. A. L., De Araújo, G. H., Schwarz, M. F., Farias, F. N. N., Bondan, L., Granville, L. Z., and Abelém, A. J. G. (2025). Advancing open ran deployment and management on the openran@brasil testbed. In *NOMS 2025-2025 IEEE Network Operations and Management Symposium*, pages 1–3.
- Guemdani, M., Gu, P., Phung, C. D., and Secci, S. (2025). Comparative E2E performance analysis of O-RAN designs in a 5G standalone testbed. In *Proceedings of the 26th IEEE International Conference on High Performance Switching and Routing (HPSR)*, pages 1–8, Osaka, Japan. hal-05039989.
- Hassouna, S., Kaur, J., Kizilkaya, B., Kazim, J. u. R., Ansari, S., Kherani, A. A., Lall, B., Abbasi, Q. H., and Imran, M. (2025). Development of open radio access networks (O-RAN) for real-time robotic teleoperation. *Communications Engineering*, 4(1):176.
- Jere, S. et al. (2024). Sajd: Self-adaptive jamming attack detection in ai/ml integrated 5g o-ran networks. In *Proceedings of the IEEE (ArXiv Preprint)*. IEEE.
- Mekki, M. (2024). *Enabling Zero-Touch Cloud Edge Computing Continuum Management*. Doctoral dissertation, Sorbonne Université – EURECOM, Sophia Antipolis, França. Directed by Prof. Adlen Ksentini.
- Silva, P., Medeiros, L., Doria, M., Campos, A., Neto, A., and Sousa, V. (2025). Monitoramento unificado para multiplataformas e2 em sistemas open ran. In *Anais do XLIII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais (SBrT 2025)*, Natal, RN, Brasil. SBrT.