
# Hórus-CDS: O Olho Digital na Segurança das Smart Grids

**Alexandro de O. Paula[1], Weslei F. Santos[2], Vinícius P. Gonçalves[1], André L. M. Serrano[1], Rodolfo I. Meneguette[3], Roger Immich[4], Geraldo P. Rocha Filho[2]**

[1]Universidade de Brasília (UnB)

[2]Universidade Estadual do Sudoeste da Bahia (UESB)

[3]Universidade de São Paulo (USP)

[4]Universidade Federal do Rio Grande do Norte (UFRN)

alexandro.paula@aluno.unb.br, 202011393@uesb.edu.br, vpgvinicius@unb.br

andrelms@unb.br, meneguette@icmc.usp.br, roger@imd.ufrn.br

geraldo.rocha@uesb.edu.br

***Abstract.*** *This work proposes the Hórus-Cyber Detection for Smart Grids (Hórus-CDS), a tool based on temporal neural networks for detecting anomalies in the network traffic of a Smart Grid (SG). The model employs Temporal Convolutional Networks (TCNs), which leverage dilated causal convolutions to capture sequential patterns and enhance real-time inference. Unlike conventional approaches, the tool enables greater parallelization, lower latency, and an interface for monitoring and decision-making. The tool was validated in two scenarios: a simulated SG environment and a real-world deployment at Neoenergia, where its resilience against DDoS attacks was tested. The results demonstrated that Hórus-CDS outperformed other approaches, ensuring the operational continuity of power distribution systems.*

***Resumo.*** *Este trabalho propõe o Hórus-Cyber Detection for Smart Grids (Hórus-CDS), uma ferramenta baseada em redes neurais temporais para detectar anomalias no tráfego de rede de uma Smart Grids (SG). O modelo utiliza Temporal Convolutional Networks (TCNs), que exploram convoluções causais dilatadas para capturar padrões sequenciais e aprimorar a inferência em tempo real. Diferente de abordagens convencionais, a ferramenta permite maior paralelização, menor latência e uma interface para monitoramento e tomada de decisões. A ferramenta foi validada em dois cenários: uma simulação de SGs e um ambiente real da Neoenergia, em que foram testadas sua resiliência contra ataques DDoS. Os resultados demonstraram que o Hórus-CDS superou outras abordagens, garantindo a continuidade operacional dos sistemas de distribuição elétrica.*

## 1. Introdução

A transformação digital dos ativos de Tecnologia da Informação (TI) e Tecnologia da Operação (TO) impulsionou inovações no setor elétrico, levando ao desenvolvimento das *Smart Grids* (SGs). Conforme o *National Institute of Standards and Technology* (NIST) [Gopstein et al. 2021], uma SG é uma subestação elétrica moderna

que integra fluxos bidirecionais de energia e comunicação, utilizando Dispositivos Eletrônicos Inteligentes (IEDs) para otimizar a gestão de equipamentos e aumentar a eficiência [Buchholz and Styczynski 2020]. No entanto, essa evolução tecnológica ampliou a superfície de ataque, expondo vulnerabilidades que ameaçam a disponibilidade desses sistemas [Paula et al. 2022, Tatipatri and Arun 2024]. Entre os principais riscos cibernéticos estão *phishing* contra operadores, manipulação de sensores e *malware* em dispositivos inteligentes. Os ataques de Negação de Serviço Distribuído (DDoS) [MITRE ATT&CK 2024, CryptoID 2024, CNN Brasil 2022] são particularmente críticos, pois direcionam grandes volumes de requisições maliciosas a serviços essenciais, tornando-os indisponíveis e comprometendo a comunicação e o controle das infraestruturas elétricas.

O uso de *Machine Learning* (ML) surge como uma alternativa para fortalecer a segurança cibernética das SGs, permitindo a detecção de padrões anômalos e ameaças em tempo real. Modelos tradicionais de ML são utilizados para classificar o tráfego, identificar anomalias ou melhorar a representação dos dados e a acurácia na detecção de ataques [Iqbal and Pooja 2019, Martinelli et al. 2022, Ravinder and Kulkarni 2023, de Oliveira et al. 2023, Naeem et al. 2024]. No entanto, abordagens tradicionais apresentam limitações na identificação de ataques dinâmicos, como os de DDoS, devido à incapacidade de capturar dependências temporais entre eventos de rede. Para lidar com essa limitação, redes neurais temporais, como *Long Short-Term Memory* (LSTM), *Gated Recurrent Units* (GRU) e *Temporal Convolutional Networks* (TCN), são utilizadas para modelagem de séries temporais. As TCNs oferecem vantagens computacionais sobre redes recorrentes, permitindo paralelização e menor latência, tornando-as adequadas para operações em tempo real, característica presente nesta pesquisa.

Diferentes trabalhos foram propostos para a predição de ataques cibernéticos em SGs. Em [Iqbal and Pooja 2019], demonstram a viabilidade de métodos supervisionados na detecção de anomalias em dados de IEDs, enquanto [Martinelli et al. 2022] propõe mecanismos temporais aplicados a *logs* SCADA para aprimorar a precisão da detecção. Outras pesquisas focam na identificação de consumo anômalo [Ravinder and Kulkarni 2023] ou na classificação de ataques com aprendizado profundo otimizado [Naeem et al. 2024]. No entanto, essas abordagens tratam apenas parte do problema, sem refletir um monitoramento contínuo e adaptável. Estudos como [Roy and Shin 2019] analisam cenários de dados desbalanceados, enquanto [Khoei and Kaabouch 2023] explora diferentes modelos sem considerar o impacto dos ajustes de hiperparâmetros na acurácia. Além disso, esses estudos falham em integrar ML a ferramentas práticas de monitoramento e operação em tempo real, limitando sua aplicabilidade em cenários críticos. A maioria dos modelos ainda está distante de atender às demandas operacionais das SGs, reforçando a necessidade de ferramentas mais adaptativas, como a proposta investigada nesta pesquisa.

Motivado por essas lacunas, este trabalho propõe o *Hórus-Cyber Detection for Smart Grids* (Hórus-CDS), uma ferramenta desenvolvida para detectar ataques DDoS em SGs. Para tanto, o Hórus-CDS foi modelado utilizando uma TCN, permitindo capturar dependências temporais no tráfego da rede e identificar padrões anômalos com maior eficiência. Diferente de abordagens convencionais, que negligenciam a estrutura sequencial dos ataques, a TCN emprega convoluções causais e dilatadas para aprimorar a detecção, garantindo inferência rápida e reduzindo a latência. Essa característica

torna o Hórus-CDS mais adequado para cenários operacionais em tempo real, permitindo a identificação precoce de ameaças e maior precisão na resposta a incidentes. Além da detecção e mitigação de ataques, o Hórus-CDS integra uma interface interativa para monitoramento, possibilitando que operadores acompanhem a atividade do modelo em produção e tomem decisões estratégicas com base nas previsões geradas. Os experimentos realizados demonstraram a viabilidade da ferramenta em fornecer respostas rápidas e eficazes, tornando-a uma solução escalável e adaptável a novas ameaças emergentes.

O restante deste artigo está estruturado da seguinte maneira. A Seção 2 apresenta o desenvolvimento da ferramenta Hórus-CDS, enquanto que na Seção 3 apresenta o roteiro da demonstração. A Seção 4 apresentada como o Hórus-CDS foi validado. Por fim, a Seção 5 apresenta a conclusão e os trabalhos futuros.

## 2. Ferramenta Proposta

Esta seção apresenta o Hórus-CDS, uma ferramenta que permitindo capturar dependências temporais no tráfego da rede e identificar padrões anômalos de ataques cibernéticos em SGs. O Hórus-CDS emprega convoluções causais e dilatadas para aprimorar a detecção, garantindo inferência rápida e reduzindo a latência por meio de uma TCN. Ainda a ferramenta Hórus-CDS integra uma interface interativa para monitoramento, possibilitando que operadores acompanhem a atividade do modelo em produção e tomem decisões estratégicas com base nas previsões geradas.

A Figura 1 apresenta a arquitetura e o fluxo de funcionamento do Hórus-CDS. Para detectar padrões anormais nas séries temporais de uma SG, são coletados dados de *logs* em tempo real do sistema de supervisão SCADA. Esses dados refletem o comportamento dos comandos realizados no sistema de supervisão SCADA em relação aos *Intelligent Electronic Devices* (IEDs), que controlam equipamentos críticos, como disjuntores e transformadores. Esses dispositivos são essenciais para a continuidade do fornecimento de energia elétrica e frequentemente são alvos de ataques cibernéticos. Os *logs* contêm informações, como o tipo de acesso (remoto ou local), a data do acesso, o tempo de execução do comando e os padrões operacionais. Após a captura, os dados operacionais das SGs são armazenados em uma base de dados estruturada e passam por um processo de pré-processamento. Essa etapa inclui limpeza, normalização e transformação de variáveis temporais, visando aprimorar a qualidade das entradas para os modelos de aprendizado de máquina utilizados na detecção de anomalias.

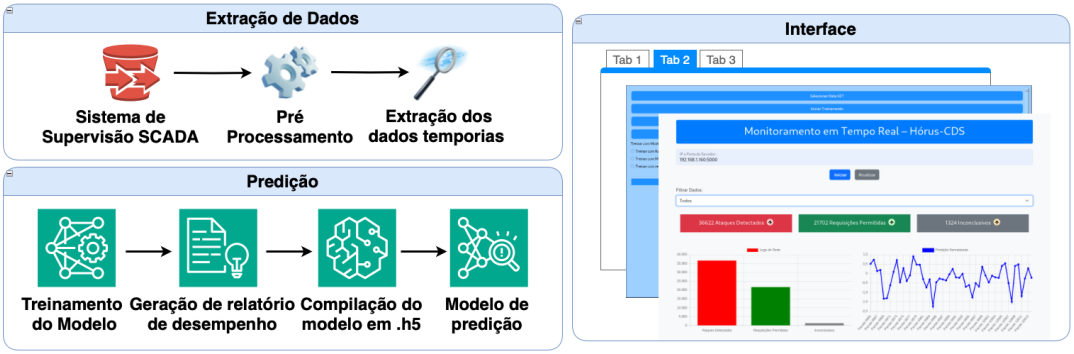


**Figura 1. Ambiente de funcionamento do Hórus-CDS.**

Após a etapa de obteção dos dados, Hórus-CDS utiliza, por padrão, TCN para identificar padrões anômalos em séries temporais. No entanto, a arquitetura da solução permite a substituição do TCN por qualquer outro algoritmo de ML. É valido salientar que para esta pesquisa foi implantando alem do TCN, os seguintes algoritmos LSTM, RNN, GRU, KNN, e ARIMA. Cada modelo é treinado para distinguir entre comandos normais e variações associadas a ataques, como picos de tráfego típicos de DoS e DDoS. No Hórus-CDS, o modelo selecionado é então compilado no formato *.h5* e disponibilizado para uso em produção. Para visualizar as informações e escolha do modelo, foi desenvolvida uma interface gráfica com PyQt5 para interação com os modelos de predição.

### 2.1. Pré-processamento e Extração de Características dos Dados para Predição

Os dados utilizados no Hórus-CDS são extraídos dos *logs* gerados pelos sistemas de supervisão das SGs, estruturados em formato tabular. Esses *logs* contêm informações críticas sobre as operações realizadas nos IEDs, incluindo a data e duração dos comandos executados (*Longtime*), o tipo de acesso (local ou remoto), além de atributos operacionais. Para realizar o pré-processamento, primeiramente, as colunas de data e hora foram convertidas para o formato `datetime`, garantindo consistência temporal. Em seguida, foram extraídas características como dia da semana, mês e hora, a partir das colunas `TXTDATE` e `TXTTIME`, permitindo a identificação de padrões sazonais.

Após essa extração, os dados foram normalizados utilizando a técnica *StandardScaler*, ajustando-os para média zero e desvio padrão unitário. Além disso, registros com valores ausentes foram tratados por interpolação linear, conforme a densidade dos dados, minimizando ruídos nos padrões que serão aprendidos pelos modelos. Para capturar tendências temporais, foram geradas variáveis derivadas, como a média móvel (`LONGTIME_MA`), calculada com uma janela deslizante de três períodos, suavizando oscilações de curto prazo. O desvio padrão (`LONGTIME_STD`), na mesma janela, para medir a dispersão dos valores ao redor da média, e auxiliar na detecção de flutuações abruptas associadas a anomalias. Além disso, foram criadas variáveis de defasagem (*lag features*) para capturar dependências temporais no histórico das operações dos IEDs. Foram geradas defasagens de 1, 2 e 3 períodos da variável alvo `LONGTIME`, garantindo que os modelos tenham acesso ao contexto recente da série temporal durante a predição.

### 2.2. Mecanismo Temporal Convolutional de Predição no Hórus-CDS

As TCNs têm sido amplamente utilizadas na modelagem de dados sequenciais devido à sua capacidade de capturar dependências temporais em larga escala com eficiência computacional superior às redes neurais recorrentes tradicionais. No Hórus-CDS, o uso de TCNs desempenha um papel fundamental na detecção de ataques DDoS em SGs, permitindo a análise de padrões de tráfego de rede de forma mais robusta e adaptativa. Ao contrário de arquiteturas recorrentes, como LSTM e GRU, que também foram avaliadas nesta pesquisa, que processam dados sequencialmente e sofrem com dificuldades de paralelização, as TCNs utilizam convoluções causais dilatadas para capturar relações temporais sem comprometer a eficiência computacional. Essa abordagem permite que o modelo aprenda padrões complexos de tráfego de rede e identifique anomalias sem a necessidade de manter estados internos como ocorre nas redes recorrentes.

A arquitetura do TCN empregada no Hórus-CDS é composta por múltiplas camadas convolucionais dilatadas, permitindo um campo receptivo expandido ao longo da

sequência temporal no sistema de supervisão SCADA. Essa configuração possibilitou que o modelo identifica-se padrões de ataque, como o aumento repentino de requisições maliciosas características de DDoS, mesmo quando dispersos no tempo. Além disso, camadas de normalização e *dropout* foram incorporadas para mitigar problemas de *overfitting* e estabilizar o aprendizado.

A implementação do TCN no Hórus-CDS segue princípios fundamentais que otimizam a detecção de anomalias em tráfegos de rede de SGs. Primeiramente, as convoluções causais dilatadas garantem que a saída de cada neurônio seja influenciada apenas por entradas passadas, respeitando a causalidade do tráfego de rede. Além disso, a paralelização eficiente permitida pelas convoluções, em contraste com as redes recorrentes, reduz significativamente o tempo de inferência do modelo. O uso de múltiplos filtros convolucionais e skip connections assegura a preservação de informações de curto e longo prazo, facilitando a detecção de variações súbitas no tráfego. Por fim, a abordagem convolucional proporciona baixa latência na inferência, possibilitando a análise de pacotes de rede em tempo real, tornando a solução adequada para cenários operacionais de SGs.

A implementação do TCN no Hórus-CDS foi estruturada para otimizar a detecção de anomalias em tráfego de redes de SGs, seguindo quatro princípios fundamentais. Primeiramente, as convoluções causais dilatadas garantem que a saída de cada neurônio seja influenciada apenas por entradas passadas, respeitando a causalidade do tráfego de rede. Além disso, a paralelização eficiente das convoluções, ao contrário das redes recorrentes, reduz significativamente o tempo de inferência do modelo. Para assegurar a preservação de padrões de curto e longo prazo, a arquitetura emprega múltiplos filtros convolucionais e *skip connections*, melhorando a identificação de variações súbitas no tráfego da SG. Por fim, a abordagem convolucional proporciona baixa latência na inferência, permitindo a análise de pacotes de rede em tempo real e tornando o Hórus-CDS adequado para aplicações operacionais em SGs.

### 2.3. Construção da Interface com o Framework PyQt5

Um dos principais componentes de visualização do Hórus-CDS é a interface gráfica (Figura 2 e Figura 3) que foi desenvolvida com o PyQt5. O PyQt5 é um framework baseado na biblioteca Qt que permite a criação de interfaces interativas para aplicações em Python. A interface do Hórus-CDS foi desenvolvida para fornecer uma experiência interativa que possibilita o gerenciamento dos modelos preditivos e o operador da SGs, a visualização dos resultados e a análise das previsões.

A interface do Hórus-CDS facilita a interação do usuário por meio de botões de controle, implementados com `QPushButton` do *PyQt5*, permitindo iniciar o treinamento, selecionar o *dataset* e interromper a execução. A escolha do modelo preditivo é feita por *radio buttons* (`QRadioButton`), associando cada opção a um modelo específico. A interface inclui uma barra de progresso (`QProgressBar`) para exibir o *status* do treinamento em tempo real e uma tabela de visualização dos *logs* (*QTableWidget*), permitindo rolagem, organização de colunas e análise prévia dos dados. Além disso, a interface realiza o monitoramento em tempo real das requisições de rede, classificando-as em ataques detectados, requisições permitidas e eventos inconclusivos. O painel exibe um resumo quantitativo dessas categorias, gráficos de tráfego e botões para iniciar ou atualizar a análise, garantindo maior controle sobre a segurança da rede.

A interface do Hórus-CDS foi implementada utilizando um layout baseado na classe `QVBoxLayout`, permitindo o empilhamento organizado dos componentes gráficos. Além disso, eventos de clique e seleção foram programados utilizando o mecanismo de sinais e slots do PyQt5, garantindo uma resposta dinâmica às interações do usuário.

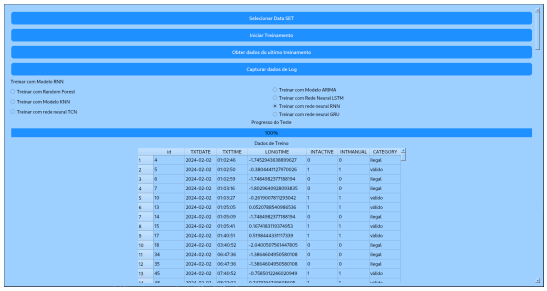

**Figura 2. Interface de configuração, Hórus-CDS**


**Figura 3. Interface de monitoramento, Hórus-CDS**

## 3. Roteiro de Demonstração

Esta seção apresenta os links para a demonstração do Hórus-CDS, incluindo o código-fonte, um vídeo explicativo e a documentação detalhada da ferramenta. A ferramenta foi desenvolvida para ser leve e otimizada, permitindo sua execução em qualquer computador que atenda a uma configuração mínima de hardware.

- Código Fonte e Documentação:
  - `https://github.com/wesleiferreira98/Horus-CDS`
- Vídeo de explicação:
  - `https://youtu.be/1mV7sXT_koc`

## 4. Avaliação de Desempenho

O Hórus-CDS foi validado em duas etapas, sendo elas:

- O Hórus-CDS foi avaliado em uma plataforma de simulação de SG, em que os IEDs representaram o nível de processos e os ativos de rede os níveis de estação e *bay*. O ambiente foi virtualizado no Fedora Server, configurado para simular ataques DDoS. A API, desenvolvida em Python, utilizou as bibliotecas `Flask`, `TensorFlow` e `scikit-learn`, e os modelos ML foram treinados e armazenados no formato `.h5` para inferência sobre tráfego de rede. A infraestrutura foi implementada no VirtualBox, com uma interface de rede em modo bridged, permitindo que a VM obtivesse um IP na mesma sub-rede da hospedeira.
- O Hórus-CDS foi implantado em um ambiente real de SG na infraestrutura da Neoenergia em Brasília (Figura 4). Esse ambiente permitiu validar a ferramenta em um cenário operacional, testando os modelos sob tráfego real, variabilidade e tentativas de ataque. A plataforma da Neoenergia seguiu normas internacionais, incluindo a IEC 61850, que organiza os IEDs em três níveis – Estação, Bay e Processo –, separados por barramentos distintos para comunicação. Para garantir interoperabilidade, foram utilizados IEDs de diferentes fabricantes e ativos de rede convencionais, incluindo um Next-Generation Firewall (NGFW), que incorpora antivírus, Sistemas de Proteção/Detecção de Intrusão (IPS/IDS), controle de aplicação e Web Application Firewall (WAF).


**Figura 4. Infraestrutura de uma SG da Neoenergia com a implantação do Hórus-CDS**

### 4.1. Avaliação em Simulação de Smart Grid

Na Figura 5, é apresentada a distribuição das requisições processadas pela ferramenta, categorizadas como ataques detectados, requisições permitidas e requisições inconclusivas. O Hórus-CDS processou um total de 43.000 requisições, das quais 20.000 foram classificadas como maliciosas, 22.000 foram identificadas como normais, e 1.000 resultaram em classificações inconclusivas. O pequeno percentual de decisões inconclusivas pode indicar cenários nos quais os padrões identificados pelo modelo não foram suficientemente discriminativos para garantir uma classificação precisa. Já a Figura 6 apresenta a análise das últimas dez requisições processadas, comparando o valor de predição com o tempo de resposta, enquanto os pontos vermelhos indicam pacotes classificados como ataques. Observa-se que variações no tempo de resposta estão correlacionadas com a identificação de tráfego anômalo, sugerindo que ataques podem impactar a latência das requisições. Requisições normais tendem a manter tempos de resposta mais estáveis, enquanto padrões suspeitos apresentam oscilações mais acentuadas. Esses resultados demonstram que o Hórus-CDS é capaz de detectar ataques cibernéticos associando padrões de tráfego a variações temporais, permitindo ajustes estratégicos para mitigar riscos e aprimorar a segurança operacional das SGs.

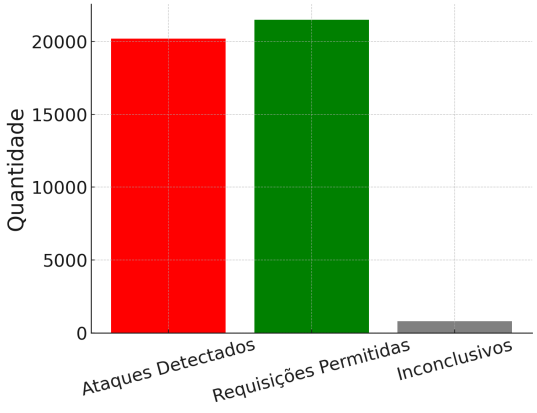


**Figura 5. Análise de requisições de rede para ataques**

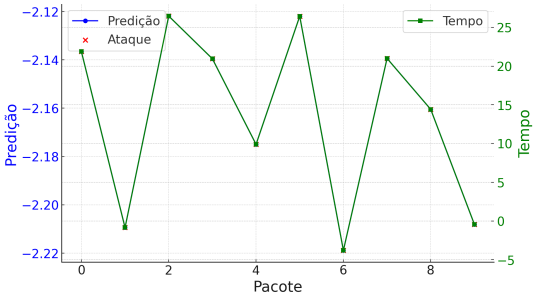


**Figura 6. Predições em função do tempo de resposta**

### 4.2. Avaliação em Ambiente Real de Smart Grid

A Tabela 1 apresenta um comparativo da disponibilidade dos disjuntores sob diferentes abordagens de defesa contra ataques DDoS em SGs. No cenário padrão, isto é, baseline que conta apenas com firewall convencional, todos os IEDs (SEL-421, ABB 620 e GE T60) foram comprometidos, resultando na abertura dos disjuntores com tempos médios variando entre 0,5 ms e 0,75 ms. A implementação do Hórus-CDS demonstrou a melhor performance, prevenindo completamente a abertura dos disjuntores em todos os IEDs, garantindo assim a continuidade operacional da infraestrutura elétrica. Outras abordagens, que foram avaliadas usando o Hórus-CDS como LSTM e GRU, mitigaram parcialmente os impactos do ataque, evitando a abertura dos disjuntores nos IEDs ABB 620 e GE T60, mas ainda apresentaram vulnerabilidade no SEL-421, em que foi registrado um tempo de abertura de 1,0 ms. Modelos baseados em KNN e Random Forest tiveram desempenho inconsistente, permitindo a abertura de alguns disjuntores em tempos variáveis. Já a solução baseada em ARIMA obteve um dos piores desempenhos, permitindo a abertura dos disjuntores em todos os IEDs, com tempos próximos aos do cenário sem mitigação.

**Tabela 1. Avaliação do Hórus-CDS na plataforma da Neoenergia**

| Solução | SEL-421 (ms) | ABB 620 (ms) | GE T60 (ms) |
|---|---|---|---|
| **Baseline** | ∼0.5 (S) | ∼0.5 (S) | ∼0.75 (S) |
| **Hórus-CDS** | **- (N)** | **- (N)** | **- (N)** |
| **LSTM** | ∼1.0 (S) | - (N) | - (N) |
| **GRU** | ∼1.0 (S) | - (N) | - (N) |
| **ARIMA** | ∼1.0 (S) | ∼0.75 (S) | ∼0.75 (S) |
| **KNN** | - (N) | ∼1.0 (S) | ∼0.5 (S) |
| **Random Forest** | - (N) | - (N) | ∼0.2 (S) |

**Legenda: S** = Disjuntor afetado — **N** = Disjuntor íntegro

## 5. Conclusão e Trabalhos Futuros

Neste trabalho, foi apresentada a ferramenta Hórus-CDS, desenvolvida para detectar ataques DDoS em SGs por meio de uma TCN. O Hórus-CDS foi projetado para identificar padrões anômalos em tráfego de rede de sistemas SCADA para mitigar ameaças como ataques de DoS e DDoS. Diferente de abordagens convencionais, a TCN utilizada no Hórus-CDS permite capturar dependências temporais com baixa latência e maior capacidade de generalização, tornando a ferramenta viável para operação em tempo real. Os experimentos realizados demonstraram a eficácia do Hórus-CDS tanto em ambiente simulado quanto em um cenário operacional real, em que foi implantado na infraestrutura da Neoenergia. Como trabalhos futuros, pretende-se expandir a avaliação da ferramenta para diferentes tipos de ataques cibernéticos, como MITM e FDI. Além disso, pretende-se implementat técnicas de *Few-shot Learning* para detectar ameaças desconhecidas, reduzindo a dependência de grandes volumes de dados rotulados.

**Agradecimento:** Os autores gostariam de agradecer o apoio financeiro da Fundação de Apoio à Pesquisa do Distrito Federal (FAPDF, processo 550/2024).

## Referências


Buchholz, B. and Styczynski, Z. (2020). *Smart Grids - Fundamentals and Technologies in Electric Power Systems of the future*. Springer.

CNN Brasil (2022). Organizações estatais da ucrânia relatam ataques cibernéticos. https://www.cnnbrasil.com.br/internacional /organizacoes-estatais-da-ucrania-relatam-ataques-ciberneticos/#: :text=A%20Naftogaz%2C %20empresa%20estatal%20de,no%20aplicativo%20de%20mensagens%20Telegram.

CryptoID (2024). Brasil: ciberataques aumentam 38% no primeiro trimestre de 2024. https://cryptoid.com.br/pesquisas-seguranca-da-informacao-e-ciberseguranca/brasil-ciberataques-aumentam-38-no-primeiro-trimestre-de-2024/.

de Oliveira, J. A., Gonçalves, V. P., Meneguette, R. I., de Sousa Jr, R. T., Guidoni, D. L., Oliveira, J. C., and Rocha Filho, G. P. (2023). F-nids—a network intrusion detection system based on federated learning. *Computer Networks*, 236:110010.

Gopstein, A., Nguyen, C., O'Fallon, C., Hastings, N., Wollman, D., et al. (2021). *NIST framework and roadmap for smart grid interoperability standards, release 4.0*. Department of Commerce. National Institute of Standards and Technology ....

Iqbal, A. and Pooja (2019). Intrusion detection in smart grid using machine learning approach. *Journal of Computational and Theoretical Nanoscience*, 16(9):3808–3816.

Khoei, T. T. and Kaabouch, N. (2023). A comparative analysis of supervised and unsupervised models for detecting attacks on the intrusion detection systems. *Information*, 14(2):103.

Martinelli, F., Mercaldo, F., and Santone, A. (2022). A method for intrusion detection in smart grid. *Procedia Computer Science*, 207:327–334. Presented at the 26th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2022).

MITRE ATT&CK (2024). Denial of service. https://attack.mitre.org/techniques/T0814/.

Naeem, H., Band, S. S., Ahmed, R., Akram, A., Bastidas-Arteaga, E., and Choo, K.-K. R. (2024). Classification of intrusion cyber-attacks in smart power grids using deep ensemble learning with metaheuristic-based optimization. *Expert Systems*, 41(1):e13003.

Paula, A. d. O., Meneguette, R. I., Giuntini, F. T., Peixoto, M. L., Gonçalves, V. P., and Rocha Filho, G. P. (2022). Strayer: A smart grid adapted automation architecture against cyberattacks. *Journal of Information Security and Applications*, 67:103195.

Ravinder, M. and Kulkarni, V. (2023). Intrusion detection in smart meters data using machine learning algorithms: A research report. *Frontiers in Energy Research*, 11:1–7.

Roy, D. D. and Shin, D. (2019). Network intrusion detection in smart grids for imbalanced attack types using machine learning models. In *ICTC 2019 - International Conference on Information and Communication Technology Convergence*, pages 576–581, Socorro, NM, USA. IEEE.

Tatipatri, N. and Arun, S. (2024). A comprehensive review on cyber-attacks in power systems: Impact analysis, detection, and cyber security. *Academic Research Velllore Institute of Technology*, 2.