IoTEdu: Network Services for the Automated Management of Campus Internet-of-Things Networks

Leonardo Bitzki[1,2], Diego Kreutz[2], Leandro Bertholdo[1]

[1]Universidade Federal do Rio Grande do Sul (UFRGS)

[2]AI Horizon Labs
Programa de Pós-Graduação em Engenharia de Software (PPGES)
Universidade Federal do Pampa (UNIPAMPA)

{bitzki,bertholdo}@ufrgs.br, kreutz@unipampa.edu.br

Abstract. *This work presents a comparative evaluation of leading integrated gateway and network services platforms, encompassing opensource, freemium, and proprietary solutions, to identify the most suitable one for the core of an automated IoT network management infrastructure in campus environments. Guided by the requirements of RNP's IoTEdu project, the study analyzes key features, performance, and integration capabilities, focusing on solutions that deliver the efficiency, scalability, and security necessary for large-scale deployment. The methodology combines a literature review, functional analysis, and experiments in a virtualized environment.*

1. Introduction

The increasing adoption of IoT devices in Higher Education and Science & Technology institutions has enabled extensive monitoring, data acquisition, and automation applications, bringing measurable operational efficiencies and financial benefits. At the same time, this expansion has introduced significant challenges related to network implementation, access control, security enforcement, and overall infrastructure management. In many cases, institutional networks either rely on slow and fragmented authorization procedures or operate wireless segments with inadequate or absent authentication, exposing the environment to operational risks and security vulnerabilities. In this context, establishing a dedicated platform for IoT network management is essential to ensure proper control, visibility, and coordination of these deployments across institutional environments.

This work evaluates the main integrated gateway and network service platforms currently available, considering their architectural approaches, functional capabilities, and respective areas of specialization. The analysis compares technical criteria aligned with institutional requirements and identifies the alternative most suitable for the automated management of IoT networks in academic environments. This evaluation was carried out in accordance with the requirements of the RNP's GT IoTEdu[1], which is developing a modular and practical solution aligned with the state of the art for IoT device networks across institutions connected to RNP.

As illustrated in Figure 1, the proposal integrates the secure onboarding of IoT devices, automated network services, advanced reactive and proactive security mechanisms,

[1] https://gt-iotedu.github.io

and federated integration for authentication and management within a single platform. This combination provides a robust foundation for the continuous and reliable operation of institutional IoT networks. The adoption of an integrated gateway and network services platform is not merely a technical requirement; it represents a strategic decision that ensures IoT-driven technological advancement occurs with security, efficiency, and reliability, allowing institutions to fully leverage these technologies without compromising the integrity of their network environments or their data.

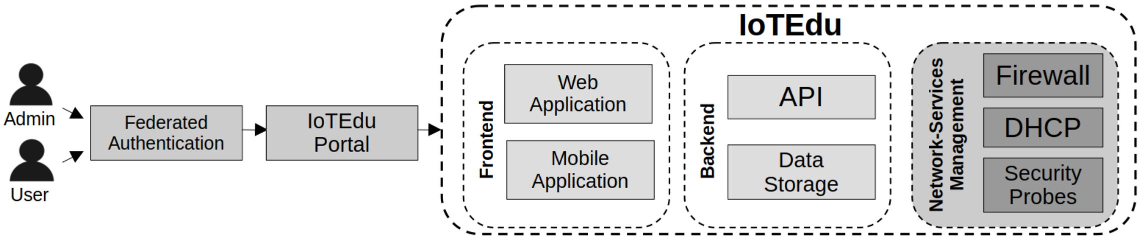

Figure 1. IoTEdu service infrastructure overview

2. Related works

This section surveys studies that individually evaluate or compare the main integrated gateway and network service platforms. A comparative view of the literature is adopted that covers multiple evaluation dimensions: *[1]* performance comparisons (based on both datasheet specification [İş 2024] and on practical tests); and *[2]* experiments in environments set up specifically for hands-on work, as presented in [Vieira et al. 2024] and [Hrițcan and Balan 2024]; *[3]* survey of functionalities, identifying and analyzing the capabilities and tools offered by the platforms, as explored in [Bin Aziz 2022]; *[4]* security considerations, examining theoretical and practical aspects related to firewalls, intrusion detection systems (IDS), and other mechanisms present in the systems, as discussed in [Che Ku et al. 2023] and [Kiratsata et al. 2022]; and *[5]* reflections on implementation costs, covering direct and indirect expenses, including equipment acquisition, licensing, and resource allocation, as exemplified in [Santos and Silva 2025]. The studies are not limited to a single dimension, and they vary in both scope and depth with which they address each topic.

Table 1. Coverage of evaluation dimensions in related work

Work	**Platform(s)**	[1]	[2]	[3]	[4]	[5]
[Bin Aziz 2022]	PacketFence	✗	✓	✓	✓	✗
[Kiratsata et al. 2022]	OPNsense and pfSense	✓	✓	✓	✓	✗
[Che Ku et al. 2023]	OPNsense	✓	✓	✓	✓	✗
[Hrițcan and Balan 2024]	pfSense	✗	✓	✓	✓	✗
[İş 2024]	pfSense, Sophos and commercial solutions	✓	✗	✓	✓	✓
[Santos and Silva 2025]	pfSense and Sophos	✗	✓	✓	✓	✓
This work	pfSense, OPNsense, PacketFence and Sophos	✓	✓	✓	✓	✓

Types of evaluation considered by the authors: [1] *Performance*; [2] *Experimental deployments*; [3] *Functionality coverage*; [4] *Security considerations*; [5] *Implementation costs.*

Table 1 summarizes the main points analyzed, making it possible to clearly see the focus of each investigation. The aim is to present a balanced overview that integrates technical evidence and practical insights, supporting informed decisions among the different platforms developed in recent years, which will be discussed in greater depth in the following sections. There is, however, a gap regarding the integrated analysis of performance, functionality, and API support in campus network scenarios with a high density of IoT devices and a strong reliance on management automation and security policies. This work specifically addresses this combination of requirements — gateway performance, API integration capability, and suitability to the modular IoTEdu architecture that this work seeks to position itself.

3. Analysis of the main platforms

For network gateway implementations that integrate critical functionalities such as firewall, DHCP server, and IDS, software-based solutions represent a flexible and economically viable alternative to proprietary hardware appliances. Among the most prominent options that natively offer the required functionalities, pfSense, OPNsense, PacketFence, and Sophos XG Firewall Home Edition were evaluated in greater depth, and it is also worth mentioning commercial software and hardware solutions such as Palo Alto VM-Series, Fortinet FortiGate, Cisco Firepower, and different RouterBOARD models from the manufacturer Mikrotik. The selection of the system most aligned with the specific operational requirements considered factors such as the degree of functional integration, development maturity, documentation quality, implementation, management and maintenance complexity, acquisition cost, and licensing model, as exemplified in Table 2.

Table 2. Comparison of basic services and licensing by tool.

Functionality	pfSense CE	OPNsense	PacketFence	Sophos XG FW
Firewall	Yes	Yes	Yes	Yes
IDS	Yes	Yes	Yes	Yes
DHCP	Yes	Yes	Yes	Yes
API	Yes	Yes	Partial	Partial
Open Source	Yes	Yes	Yes	No

3.1. pfSense (Community Edition)

pfSense Community Edition (CE) is an open-source routing, firewall, and network services platform widely used by companies and organizations of varying sizes, levels of complexity, and requirements. Originating in 2004 as a fork of the `m0n0wall` system, the project had its first official release in October 2006 and has maintained a continuous cycle of development and support ever since. It is a FreeBSD-based distribution whose architecture integrates tools and services tightly with the underlying OS, with kernel-level optimizations when appropriate to meet the platform's needs.

pfSense stands out for its flexibility and the wide range of native features it offers, representing a low-cost (or free) alternative to commercial firewalls. Furthermore, it provides a robust set of additional tools through its package system, including DHCP service (historically via ISC DHCP, and more recently via ISC Kea), Snort, Suricata, and Zeek as easily integrated IDS options, as well as a REST API that simplifies communication between external systems and its internal functions. Over nearly two decades, pfSense has established itself as a technologically mature solution, supported by an active community, extensive documentation, and optimization for environments with limited computing resources, even under intensive usage scenarios.

3.2. OPNsense

Positioning itself as a very similar alternative in terms of intended use, OPNsense emerged in 2015 from a fork of pfSense. Also based on FreeBSD, the project adopts a development philosophy that prioritizes more frequent updates and regular release cycles, with stable versions released every six months. The main differentiating efforts from the original project focus on the redesign of the management interface and the rewriting of the codebase, cited on the project's official page as the central motivation for creating the fork. Like pfSense, OPNsense offers native support for IDS using the Suricata and Zeek engines, although version availability may differ from the latest upstream releases. The base version of the operating system is also often based on a FreeBSD release slightly behind the latest stable version.

3.3. PacketFence

PacketFence is an open-source solution for wired and Wi-Fi networks, focused on device identification, authentication, and authorization (Network Access Control, NAC). It stands out for its application of policies based on user role or device type, allowing the isolation of non-compliant segments and verification of compliance requirements. It operates from passive monitoring to inline action, making it effective in containing threats in heterogeneous and large-scale environments. Installation can be done via packages for Debian or RHEL distributions, or via appliances for bare-metal environments or virtual machines. Although it is a free solution, it requires above-average minimum resources, demanding at least 4 CPU cores of 3 GHz, 16 GB of RAM, and 200 GB of storage, which can increase the cost due to the necessary hardware.

3.4. Sophos XG Firewall (Home Edition)

The Sophos XG Firewall Home Edition (HE) offers the experience of an enterprise-level commercial firewall at no cost for home use, integrating its own robust and easy-to-configure IDS that replicates the enterprise edition with a unified and refined console. Instead of independent modules, it uses a single engine that consolidates visibility and control by correlating traffic, threats, and user activity. It also offers features such as filtering, DHCP server, API, and native Active Directory integration for defining identity-based policies. Despite these advantages, it has important limitations, such as the restrictions of a license intended exclusively for home use, the lack of access to the source code, and the potential cost of migrating to a paid license if future needs exceed the limits established for the free version.

4. Experimental performance evaluation

Three virtual servers were created in a virtualized environment with Proxmox Virtual Environment: a pfSense 2.8 server, an OPNsense 24.7 server, and an Ubuntu 24.04 server with *nftables* and packet forwarding enabled. All were configured with the same computing resources, consisting of four vCPU cores of an AMD EPYC 7413 processor, 2 GB of RAM, and two 10 Gbps paravirtualized VirtIO interfaces, one interface named WAN, with access to the public network, and another named LAN, placed in a private network.

To ensure a fair comparison between the platforms, only the minimum features required for packet forwarding between the WAN and LAN interfaces were kept, with identical firewall rules and NAT configurations across all systems. Additional services with a greater impact on performance, such as IDS/IPS, proxy, and VPN, were disabled on all virtual machines during the experiments, so as to isolate the behavior of the gateway data plane. No swapping activity was observed during the experiments, ensuring that RAM was sufficient and did not bias the measurements.

Previous experiments, such as those demonstrated by [Vieira et al. 2024], indicate that OPNsense uses approximately 50% more RAM and has CPU usage similar to that of pfSense. Our measurements corroborated these findings, and to strengthen the conclusions about efficiency in resource-constrained environments, CPU and memory usage metrics of the virtual machines were also collected during the execution of the tests. Consumption was monitored by the Proxmox hypervisor every second, recording the average idle value and the peak CPU (%) and RAM (MB) utilization in each combination of platform and test.

The throughput tests consisted of sequentially running 10 rounds of the client and server components of the `NDT Measurement Lab` and `iperf` tools, in order to obtain an individual average throughput value per platform. Traffic was generated unidirectionally from WAN to LAN at maximum achievable throughput, thereby ensuring measurement consistency for gateway forwarding capacity. For the latency evaluation, 10 rounds of sending 10,000 TCP packets on port 80 were performed using the `hping3` client, in addition to sending the same volume of ICMP packets using the `fping` tool, both executed without burst limit restrictions. For each combination of platform and tool, the average of the ten runs was calculated.

In default installations and with minimal adaptations of the operating system to the virtualized environment, the throughput and latency results indicated very similar performance between pfSense and OPNsense, whereas the Ubuntu-based server showed lower performance. Table 3 presents the average throughput and latency values, as well as summarizing the average use of computing resources during the tests. The measurements of CPU and RAM usage reinforced this trend, showing that, for the evaluated scenario, pfSense achieves a good balance between performance and resource consumption, while OPNsense tends to require more memory, which is consistent with [Vieira et al. 2024]. On the other hand, the use of synthetic traffic traces (based on TCP/ICMP) and a limited number of flows does not fully represent typical IoT workloads with multiple concurrent devices, which is acknowledged as a limitation of this experimental methodology.

5. Analysis of the platforms in the context of the IoTEdu project

While pfSense and OPNsense stand out as general-purpose gateway platforms, highly customizable and with excellent performance, PacketFence is presented as a solution specialized in NAC and *captive portal*, with significantly higher resource requirements. The performance and CPU and memory usage results obtained in Section 4 reinforce that,

Table 3. Data throughput, latency, and computing resource usage tests

Test performed	pfSense CE	OPNsense	Ubuntu Server
NDT M-Lab (in Mb/s)	2811	2496	1007
iperf (in Mb/s)	2937	2641	1193
hping3 (port 80 response, in ms)	1.831	1.819	2.117
fping (average rtt, in ms)	0.181	0.179	0.259
CPU Load (idle average, in %)	0.3	0.3	0.5
CPU Load (peak, in %)	4.5	4.4	37.8
RAM Usage (idle average, in MB)	467.5	887.8	566.1
RAM Usage (peak, in MB)	511.2	976	582.5

in the evaluated scenario, pfSense offers a more favorable balance between throughput, latency, and resource consumption, a relevant characteristic for deployments in institutional environments with heterogeneous and potentially limited infrastructures. Sophos XG (HE), in turn, offers an experience more oriented toward home use, not fully meeting corporate and institutional demands, unlike the other evaluated platforms.

Since IoTEdu's architecture is modular, extensible, and non-monolithic, all interactions between the frontend, backend, gateway, security probes, and other system components must occur through API-based communication. Consequently, comprehensive API support for managing gateway functionalities and network services is a foundational requirement for the project. As shown in Table 4, the direct comparison of real-time API manipulation capabilities reveals that pfSense CE, OPNsense and PacketFence offer similar and robust support, while Sophos XG (HE) present relevant limitations. These limitations stem from the absence of API endpoints necessary for full configuration of certain functionalities, resulting in only partial or restricted control over specific services.

Table 4. Support for real-time management of features via API

Feature	pfSense CE	OPNsense	PacketFence	Sophos XG
Firewall: Aliases	Full	Full	Full	Partial
Firewall: Rules	Full	Full	Full	Partial
Firewall: KillState	Full	Full	Full	Partial
ARP Table	Full	Full	Full	Partial
DHCP IPv4 and IPv6	Full	Full	Full	Full
DHCP: Reservation	Full	Full	Full	Partial
DHCP: Leases	Full	Full	Full	Full
Logs: DHCP	Full	Full	Full	Partial
Logs: System	Full	Full	Full	Partial

It is important to clarify that this work focuses specifically on the role of edge gateways and firewalls that interconnect IoT networks with the institutional backbone, rather than on embedded IoT gateways designed for severely resource-constrained devices. The evaluated platforms are intended to run on general-purpose virtualized or bare-metal servers and function as centralized enforcement points for traffic originating from campus IoT deployments. In this sense, the study complements prior work centered on application-level protocols or embedded device gateways by addressing the network-service and security layers that sustain the operation, control, and protection of large-scale IoT environments.

6. Final considerations and future work

This work presented a comparative analysis of different integrated gateway and network service platforms to identify the most suitable alternative to the modular architecture proposed by the IoTEdu project of RNP. The evaluation considered functionalities, performance, licensing, and integration via API, aligned with the heterogeneous demands of institutional infrastructures. The results indicate that pfSense provides a strong balance between performance, flexibility, and integration, more consistently meeting the requirements of IoTEdu. OPNsense, although it shows similar performance, demonstrates comparatively lower maturity in institutional adoption and a smaller support ecosystem, while PacketFence and Sophos XG Firewall proved to be more restrictive due to limitations in licensing, access to source code, and APIs.

As future work, we intend to expand the experimental tests with concurrent loads and multiple IoT devices, evaluating the stability and scalability of the gateway in scenarios closer to production environments. We will emulate traffic patterns typical of IoT deployments (such as MQTT and CoAP), include resource-constrained devices, and vary the number of flows and simultaneous connections. This expanded methodology will address current limitations and guide the refinement of the IoTEdu deployment guidelines.

Acknowledgment. This research received partial support from the RNP[2], through the IoTEdu Working Group, and from the CAPES[3], under Financing Code 88887.954253/2024-00 and 001.

References

Bin Aziz, M. A. S. (2022). Automated host verification and authentication for iot devices using network access authentication model. In *ICCR*, pages 1–7.

Che Ku, M. R. S. A., Che Ku Nur, S. A., and Mohamad Usop, N. S. (2023). Towards secure local area network (LAN) using OPNsense firewall. *Malaysian JCAM*, 6(1).

Hrițcan, D.-F. and Balan, D. (2024). Using tailscale and pfsense for security and anonymity of iot environments. In *DAS*, pages 91–94.

Kiratsata, H. J., Raval, D. P., Viras, P. K., Lalwani, P., Patel, H., and D., P. S. (2022). Behaviour analysis of open-source firewalls under security crisis. In *WiSPNET*.

Santos, R. C. d. and Silva, V. B. d. (2025). Análise da evolução de soluções de firewall em uma instituição de saúde: um estudo de caso entre 2019 e 2023. *Revista de Gestão e Secretariado*, 16(7):e5071.

Vieira, E., Wendler, E., Rizzetti, T., and Azevedo, R. (2024). Estudo comparativo de firewalls de código aberto baseados em FreeBSD. In *Anais da XXI ERRC*. SBC.

İş, H. (2024). A comprehensive analysis of ngfws for cyber-physical system security after the crowdstrike incident. In *2024 Global Energy Conference (GEC)*, pages 12–20.

[2] `https://www.rnp.br`

[3] `https://www.gov.br/capes/pt-br`