Performance Evaluation and Threat Mitigation in Large-scale 5G Core Deployment

Rodrigo Moreira[1], **Larissa F. Rodrigues Moreira**[1], **Flávio de Oliveira Silva**[2]

[1]Institute of Exact and Technological Sciences – Federal University of Viçosa (UFV)
Rio Paranaíba – MG – Brazil

[2]Department of Informatics – School of Engineering
University of Minho (UMinho) – Braga – Portugal

{rodrigo, larissa.f.rodrigues}@ufv.br, flavio@di.uminho.pt,

Abstract. *The deployment of large-scale software-based 5G core functions presents significant challenges due to their reliance on optimized and intelligent resource provisioning for their services. Many studies have focused on analyzing the impact of resource allocation for complex deployments using mathematical models, queue theories, or even Artificial Intelligence (AI). This paper elucidates the effects of chaotic workloads, generated by Distributed Denial of Service (DDoS) on different Network Functions (NFs) on User Equipment registration performance. Our findings highlight the necessity of diverse resource profiles to ensure Service-Level Agreement (SLA) compliance in large-scale 5G core deployments. Additionally, our analysis of packet capture approaches demonstrates the potential of kernel-based monitoring for scalable security threat defense. Finally, our empirical evaluation provides insights into the effective deployment of 5G NFs in complex scenarios.*

1. Introduction

The Beyond Fifth Generation (B5G) network is designed to offer more than connectivity, enabling disruptive applications like human-machine interaction and smart environments [Prasad Tera et al. 2025]. B5G networks will drive technological transformation, meeting quantitative demands—non-terrestrial networks, higher cell density, Artificial Intelligence (AI)-based optimization, and spectrum sharing—alongside qualitative needs like scalability, sustainability, trust, human-centeredness, and digital inclusion [Tsekenis et al. 2024, Brinton et al. 2025].

Software-based B5G cores enable the community to propose and assess both evolutionary and disruptive approaches for the control and data planes within Fifth-generation of Mobile Telecommunications Technology (5G) cores, whether open-source or commercial [Rouili et al. 2024]. Many interventions, such as AI-driven security, focus on specific Network Functions (NFs) [Martins et al. 2023, Moreira et al. 2024, Suomalainen et al. 2025]. While existing studies evaluate 5G cores from various perspectives, the impact of individual NFs on user perception and host systems under chaos remains unexplored [Mukute et al. 2024].

This paper examines the impact of resource constraints on 5G NFs on User Equipment (UE) perception during registration and Protocol Data Unit (PDU) session establishment. Using a chaos engineering-based method, we apply synthetic workloads to simulate an Distributed Denial of Service (DDoS) attached to each Network Function (NF), whereas a UE sensor collects registration statistics. We statistically assessed their effects by asynchronously stressing Central Processing Unit (CPU), memory, and both. We also analyze the overhead of a container and the Machine Learning (ML)-based threat defense proposed mechanism on the host system, considering user and kernel space sniffing.

The structure of this work is as follows: Section 2 discusses methodologies analogous to the one proposed herein. Section 3 delineates the evaluation methodology. The evaluation results are presented in Section 4, and the study concludes with Section 5.

2. Related Work

Research on 5G networks has encompassed diverse aspects, from evaluating simulation and emulation tools to optimizing core network functionalities. [Rouili et al. 2024] analyzed open-source 5G RAN tools, comparing Software Defined Radios (SDR)-based, emulation, and simulation scenarios in terms of throughput, latency, resource use, coverage, and power consumption.

[Phan et al. 2024] developed two software-based 5G Core Network testbeds (one using desktop PCs and another a high-performance server), integrating container orchestration tools to enable scalable and efficient deployment. However, the authors did not consider evaluation metrics like connectivity tests, latency, and packet loss. [Bolla et al. 2024] addressed the challenge of energy efficiency in virtualized 5G User-Plane Functions (UPFs) by comparing packetization methods and conducting experiments in diverse traffic scenarios, measuring performance (CPU usage and memory usage) and energy consumption.

[Mukute et al. 2024] and [Chen and Huang 2024] investigated performance challenges in open-source 5G Core Networks, particularly free5GC, focusing on control plane functions and scalability, considering metrics such as CPU utilization and registration time. [Mukute et al. 2024] introduced a benchmarking framework linking system call optimizations to improved user registration performance, while [Chen and Huang 2024] examined scalability under varying registration scales and duplicate requests.

In contrast to the aforementioned studies, this study advances the state-of-the-art by addressing the deployment challenges of 5G core functions under chaotic workloads such as DDoS attacks. By analyzing their impact on NFs and UE registration performance, as well as demonstrating the potential of kernel-based monitoring for scalable security threat defense and the necessity of tailored resource provisioning for Service-Level Agreement (SLA) compliance.

3. Evaluation Method

To evaluate whether 5G NFs influences and under which workload circumstances impact the UE registration experience, we propose a benchmarking method based on chaos engineering with factor analysis. Figure 1 illustrates the proposed method.

Step ❶ injects workload parallel to a UE application requesting registration and session establishment PDU at a constant rate to the core. These requests assume prior UE registration in the 5G core database. The UE registration sensor application extends my5G-RANTester [Silveira et al. 2022], with adaptations ensuring a constant duration and rate of requests to extract experimental latencies.

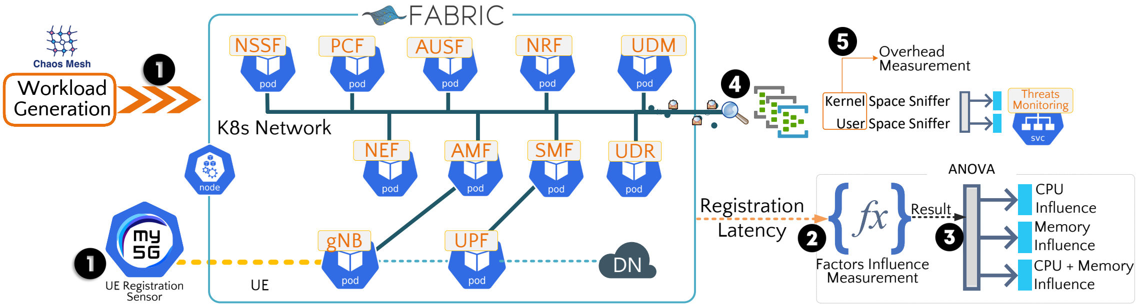


Figure 1. Benchmarking Method.

For workload injection into 5G core components, we used Chaos Mesh, a chaos engineering tool for inducing stress in Kubernetes clusters. Table 1 details the stressed components, load intensity, and time parameters customizable and empirically defined in our experiments.

Table 1. Stress Test Scenarios and Parameters.

Scenario	CPU Load (%)	Memory (MiB)	Duration (s)
CPU Stress	50	-	20
Memory Stress	-	512	20
CPU + Memory Stress	50	512	20

Step ❷ merges sensor application latency data with the timestamps of each load-injection profile in the 5G core pods. Our deterministic workload induction method records start and end timestamps, enabling correlation between induced load periods and the UE registration latency.

Step ❸ evaluates the performance impact of induced load on 5G core components using Analysis of Variance (ANOVA). We assess the influence of each NF and stress pattern on resources affecting UE registration and PDU session establishment. Three factors are considered: CPU, Memory, and CPU + Memory, varied as per Table 1. Their effects on UE registration time were analyzed using the Linear Mixed Model (LMM) method.

Step ❹ employs packet sniffing on the container interface to feed an ML model that classifies packets as attacks or normal traffic. Packets are captured using (i) user-space tools (e.g., tcpdump, Wireshark) or (ii) kernel-space methods (Extended Berkeley Packet Filter (eBPF)). We measure CPU and memory usage to assess the overhead of each approach. In Step ❺ , discrete statistical methods analyze this impact to determine the most suitable option for production.

4. Results and Discussion

We conducted experiments on the Fabric testbed [Baldin et al. 2019], deploying a VM with 16 vCPUs, 32 GB of RAM, a single-node Kubernetes cluster, Chaos Mesh (a chaos engineering tool), free5GC, and NetData (for monitoring the entire infrastructure). This setup identified the most load-sensitive 5G core NF and its impact on UE registration time. Additionally, we evaluated two packet-sniffing methods in containerized environments to assess their monitoring overhead.

For packet sniffing, we used kernel-space (`ptcpdump`) and user-space (`Ksniff`) methods, which differ in Kubernetes pod monitoring. `Ptcpdump` leverages eBPF to capture and annotate packets in the kernel. In contrast, `Ksniff` operates as a `Kubectl` plugin, uploading a compiled `tcpdump` binary to the target pod and redirecting its output for analysis.

4.1. Impact of Attacks on NF

We first measured how a DDoS attack compromises a 5G NF to assess its impact on UE registration and PDU session establishment. Using concurrent goroutines in Go, we generated high-volume Hypertext Transfer Protocol (HTTP) requests to overload the Access and Mobility Management Function (AMF), simulating a DDoS attack on its endpoint.

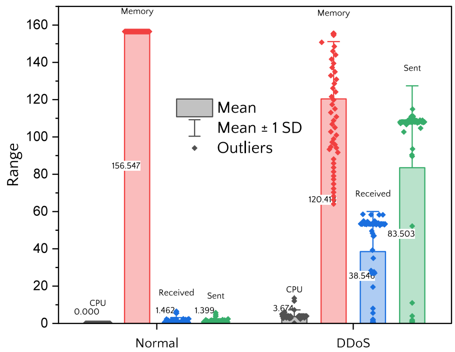


Figure 2. Influence of Attacks on AMF.

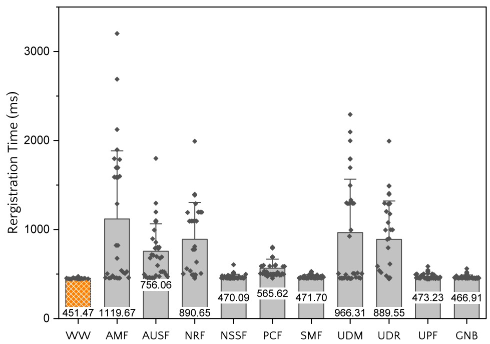
Figure 2 illustrates the impact of a DDoS attack on the AMF. During the attack, CPU utilization spiked to a mean of 3.674, while memory usage dropped to 120.412, indicating disruptions. Network traffic surged, with received and sent data increasing to 38.546 and 83.503, respectively. These results highlight the strain on AMF resources, affecting both efficiency and stability, emphasizing the need for advanced detection and mitigation. Additionally, 5G NF workloads can degrade service and impact UE quality perception, as discussed in the following subsections.

4.2. Variability Analyses

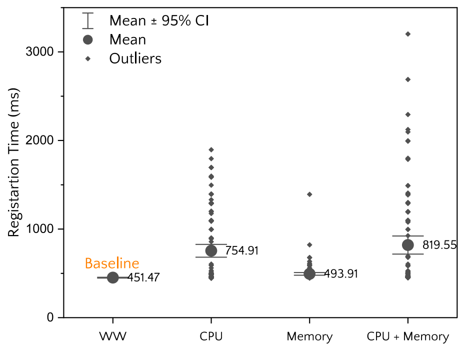
Assuming DDoS attacks degrade NFs performance, we used a workload simulator to assess their impact on UE registration. The analysis examined workload-induced variability in open-source 5G core NFs. As shown in Fig. 3a, AMF was the most affected, significantly impacting UE registration time and highlighting its critical role under stress.

As shown in Fig. 3b, the CPU + Memory workload has the higher impact on UE registration time in the open-source 5G core. Thus, in microservices-based 5G deployments, varying CPU and Memory demands across NFs must be considered for production environments.

After identifying the variability of the registration time considering the NF and the type of workload, we sought to determine whether the NF influences the response time as a random effect or whether different NFs can affect the registration time, thus being a co-variate in the analysis. To make this decision, a test was conducted to verify whether there was a significant difference between the average times for NFs.



(a) Registration Time considering different NFs.



(b) Induction of different workloads.

Figure 3. Comparison of variability by NF, workloads, and the baseline.

We conducted an ANOVA test to assess the impact of NF on Registration Time. The results (Table 2) show that NF is highly significant ($p < 0.001$), with an F-statistic of 18.611059. This confirms that NF workload influences UE registration as a random effect.

Table 2. ANOVA Results for NF Influence on Registration Time.

Source	Sum of Squares (SS)	Degrees of Freedom (df)	F-Statistic (F)	p-Value (p)
C(NF)	1.972265×10^7	9	18.611059	1.352433×10^{-25}
Residual	4.356657×10^7	370	-	-

4.3. Influence Analyses

The third part of our experiment examines the effect of each NF on registration time using a three-way ANOVA, with NF as a random effect in the LMM. We analyzed CPU, Memory, and CPU+Memory workloads to assess their impact. With NF showing significance in prior analysis, we included LMM as a random effect. The baseline registration time without workload injection was 451.46 ms (standard deviation: 7.80 ms). The LMM results are summarized below:

Table 3. Mixed Linear Model Regression Results.

Source	Coefficient (β)	Std. Error	z-value	p-value	95% CI
Intercept	451.467	270.491	1.669	0.095	[-78.685, 981.619]
C(stress_test)[T.CPU]	352.239	283.976	1.240	0.215	[-204.344, 908.821]
C(stress_test)[T.Memory]	47.211	283.729	0.166	0.868	[-508.886, 603.309]
C(stress_test)[T.CPU+Memory]	420.092	284.012	1.479	0.139	[-136.561, 976.745]
Group Var	67402.169	113.331	-	-	

The intercept represents the average registration time without workload injection (451.467 ms). The CPU stress test increases registration time by 352.239 ms ($p = 0.215$), memory stress by 47.211 ms ($p = 0.868$), and their combination by 420.092 ms ($p = 0.139$), but none are statistically significant. The group variance (67402.169) indicates substantial variability among the NFs.

The LMM results show that while stress conditions do not significantly impact registration time, the combined CPU and memory stress condition (`C(stress_test)[T.CPU+Memory]`) has the largest effect. The variance associated with NF ($\hat{\sigma}_u^2 = 67402.169$) contributes considerably to the total variability, highlighting the importance of accounting for NF as a random effect.

From Figure 4, we observe that AMF has the greatest influence on the registration time, followed by Unified Data Management (UDM) and Unified Data Repository (UDR). This influence is more pronounced when the CPU and memory factors are combined, implying greater difficulty in handling the UE registration workload. For all NFs, memory stress alone does not have a relevant impact on UE registration. Therefore, in large-scale deployments, it is crucial to consider the differences in resource demands of each NF.

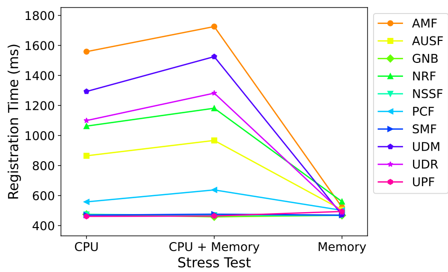
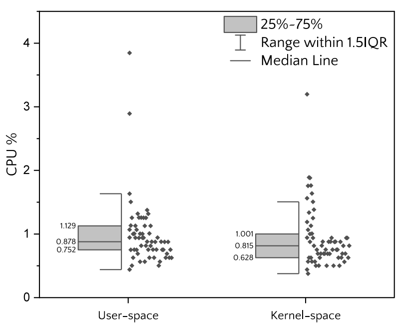


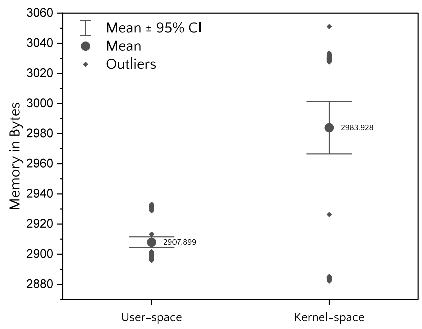
Figure 4. Workload interaction on NFs.

4.4. Monitoring Overhead

To deploy a threat defense mechanism against service degradation perceived by the UE, we propose monitoring packet exchanges between NFs by sniffing packets and transmitting their hexadecimal representation to an ML-based service for classification as normal or DDoS. This method introduces additional workload to the cluster, which we analyze in detail.



(a) CPU



(b) Memory.

Figure 5. Comparison of packet sniffing methods.

The analysis of CPU consumption (Figure 5a) revealed that the kernel-based method using eBPF had slightly lower CPU usage than the user-space method with Ksniff. While the median CPU consumption was similar (0.878% for user-space and 0.815% for kernel), the kernel-based approach showed greater amplitude in the third quartile and less data dispersion, indicating eBPF efficiently captures packets within the kernel.

The memory usage analysis (Figure 5b) indicates that the kernel-based method has a higher mean memory consumption (2983.928 bytes) compared to the user-space approach (2907.899 bytes), implying greater memory overhead. Additionally, both methods exhibit outliers, with higher dispersion in kernel-space data, indicating variability in memory performance. These sniffed packets could support an ML-based security threat defense, leveraging the kernel-space approach to enhance security in such deployments.

Using a single virtual machine simplifies deployment but limits the realism of a distributed 5G core. Similarly, synthetic traffic lacks real-world variability, which future work aims to address.

5. Concluding Remarks

This paper examines the impact of varying loads on 5G control plane NFs and their effect on user perception. While prior methodologies assessed different 5G open source cores, they did not quantify the latency impact of each NF on UE registration and PDU session establishment. This research offers quantifications to inform large-scale deployments, enabling differential resource allocation for NFs in resource-constrained environments, particularly under low-power policies.

The findings indicate that the AMF entity requires the most resources due to its significant impact on UE registration time. Kernel-based monitoring approaches like eBPF can improve security threat detection while maintaining efficiency in resource-constrained 5G deployments. Future research will assess failure resilience in open-source 5G cores, providing a reference for large-scale deployments that meet the resource needs of each NF.

Acknowledgments

We acknowledge the financial support of the FAPESP MCTIC/CGI Research project 2018/23097-3 and FAPEMIG (Grant APQ00923-24). We also thank FCT—Fundação para a Ciência e Tecnologia within the RD Unit Project Scope UID/00319/Centro ALGORITMI (ALGORITMI/UM) for partially supporting this work.

References

Baldin, I., Nikolich, A., Griffioen, J., Monga, I. I. S., Wang, K.-C., Lehman, T., and Ruth, P. (2019). Fabric: A national-scale programmable experimental network infrastructure. *IEEE Internet Computing*, 23(6):38–47.

Bolla, R., Bruschi, R., Davoli, F., Ghiorzo, P., Ivaldi, L., Lombardo, C., and Siccardi, B. (2024). Comparison of the Performance and Energy Efficiency Evaluation of 5G User-Plane Functions. In *2024 IEEE International Conferences on Internet of Things (iThings) and IEEE Green Computing & Communications (GreenCom) and IEEE Cyber, Physical & Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics*, pages 361–368.

Brinton, C. G., Chiang, M., Kim, K. T., Love, D. J., Beesley, M., Repeta, M., Roese, J., Beming, P., Ekudden, E., Li, C., Wu, G., Batra, N., Ghosh, A., Ziegler, V., Ji, T., Prakash, R., and Smee, J. (2025). Key Focus Areas and Enabling Technologies for 6G. *IEEE Communications Magazine*, 63(3):84–91.

Chen, P.-Y. and Huang, C.-Y. (2024). Evaluation of UE Registration Performance in Open-Source 5G Core Networks. In *2024 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)*, pages 1–5.

Martins, J. S. B., Carvalho, T. C., Moreira, R., Both, C. B., Donatti, A., Corrêa, J. H., Suruagy, J. A., Corrêa, S. L., Abelem, A. J. G., Ribeiro, M. R. N., Nogueira, J.-m. S., Magalhães, L. C. S., Wickboldt, J., Ferreto, T. C., Mello, R., Pasquini, R., Schwarz, M., Sampaio, L. N., Macedo, D. F., De Rezende, J. F., Cardoso, K. V., and De Oliveira Silva, F. (2023). Enhancing Network Slicing Architectures With Machine Learning, Security, Sustainability and Experimental Networks Integration. *IEEE Access*, 11:69144–69163.

Moreira, R., Moreira, L., Carvalho, T., and Silva, F. (2024). Resource Allocation Influence on Application Performance in Sliced Testbeds. In *Anais do XV Workshop de Pesquisa Experimental da Internet do Futuro*, pages 8–14, Porto Alegre, RS, Brasil. SBC.

Mukute, T., Mamushiane, L., Lysko, A. A., Modroiu, E.-R., Magedanz, T., and Mwangama, J. (2024). Control Plane Performance Benchmarking and Feature Analysis of Popular Open-Source 5G Core Networks: OpenAirInterface, Open5GS, and free5GC. *IEEE Access*, 12:113336–113360.

Phan, L.-A., Pesch, D., Roedig, U., and Sreenan, C. J. (2024). Building a 5G Core Network Testbed: Open-Source Solutions, Lessons Learned, and Research Directions. In *2024 International Conference on Information Networking (ICOIN)*, pages 641–646.

Prasad Tera, S., Chinthaginjala, R., Pau, G., and Hoon Kim, T. (2025). Toward 6G: An Overview of the Next Generation of Intelligent Network Connectivity. *IEEE Access*, 13:925–961.

Rouili, M., Saha, N., Golkarifard, M., Zangooei, M., Boutaba, R., Onur, E., and Saleh, A. (2024). Evaluating Open-Source 5G SA Testbeds: Unveiling Performance Disparities in RAN Scenarios. In *NOMS 2024-2024 IEEE Network Operations and Management Symposium*, pages 1–6.

Silveira, L. B., de Resende, H. C., Both, C. B., Marquez-Barja, J. M., Silvestre, B., and Cardoso, K. V. (2022). Tutorial on communication between access networks and the 5G core. *Computer Networks*, 216:109301.

Suomalainen, J., Ahmad, I., Shajan, A., and Savunen, T. (2025). Cybersecurity for tactical 6G networks: Threats, architecture, and intelligence. *Future Generation Computer Systems*, 162:107500.

Tsekenis, V., Barmpounakis, S., Demestichas, P., Wänstedt, S., Habibi, M. A., Schotten, H. D., Akgul, O. U., Hellaoui, H., Kousaridas, A., Zivkovic, M., Botsinis, P., Eldessoki, S., Groshev, M., and Palenius, T. (2024). Towards Beyond Communication 6G Networks: Status and Challenges. In *2024 IEEE 35th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1–6.