

B5G Core Network Openness: A Proof of Concept Implementation of Location Reporting via Network Exposure Function

Vasilis Pitsilis, Panagiotis Pavlidis, Stefanos Plastras, George Makropoulos, Andreas Sakellaropoulos, Harilaos Koumaras, Spyridon Georgoulas
Institute of Informatics and Telecommunications, NCSR Demokritos, Athens, Greece
{vpitsilis, p.pavlidis, s.plastras, gmakropoulos, asakellaropoulos, koumaras, spygeorgoulas, }@iit.demokritos.gr

Abstract—The Network Exposure Function (NEF) is a key component of the 5G and Beyond 5G (B5G) network, providing standardized access for third-party applications to network data and capabilities. Among these capabilities is the Event Monitoring API, which allows external applications to subscribe to specific network events. One such event is the reporting of user equipment (UE) location changes, enabling services that rely on real-time location awareness. In this context, this paper presents the design and implementation of an open-source, cloud-based NEF Event Monitoring Application Programming Interface (API) that adheres to relevant 3rd Generation Partnership Project (3GPP) standards. The proposed architecture minimizes integration complexity and provides a flexible framework for efficient deployment and utilization of future 6G location services, fostering innovation in intelligent environments and enabling a wide range of third-party applications.

Index Terms—5G, 6G, Network Exposure Function (NEF), UE Location, Open-Source, 3GPP, Cloud-Native, APIs.

I. INTRODUCTION

The accelerated development of mobile communication networks has fundamentally transformed how services and connections are established. Innovations such as network slicing, edge computing, and enhanced network exposure capabilities are among the revolutionary advances introduced by 5G cellular networks. One key component in this ecosystem is the Network Exposure Function (NEF), which plays a critical role by enabling third-party applications to access network features and events in a standardized and secure manner. Through its northbound Application Programming Interfaces (APIs), NEF makes the 5G network more accessible, controllable, and programmable, allowing service providers to enhance their applications by leveraging the rich capabilities offered by 5G systems [1].

Among these capabilities, real-time monitoring and retrieval of user location data are considered to be important features for enabling advanced services and applications. Efficient location tracking not only improves user experience but also enables the development of context-aware applications that depend on precise spatial information. As we transition from fifth- to sixth-generation networks, the demand for more accurate, flexible, and scalable location tracking systems is becoming increasingly critical. Such capabilities underpin a broad array

of emerging services, including smart city infrastructure, augmented reality navigation, and industrial automation.

To effectively expose 5G services, ensuring flexibility and compliance between NEF and the 5G Core (5GC) is paramount [2]. NEF interacts with multiple 5GC functions via southbound interfaces within the Service-Based Architecture (SBA). For example, NEF interfaces with the Policy Control Function (PCF) to support dynamic policy enforcement, the Network Data Analytics Function (NWDAF) to provide analytics for more informed decision-making by external applications, and the Session Management Function (SMF) to influence traffic steering towards edge servers. Crucially, the Access and Mobility Management Function (AMF) plays a central role in NEF's operation, especially for location tracking and mobility-related services [3], [4]. This close relationship with the AMF is fundamental for enabling NEF to provide accurate, real-time location information while maintaining secure and efficient network operations. Additionally, 5GC capabilities must be securely exposed through NEF, which abstracts and hides the underlying network topology to maintain security [5].

In light of the above, this paper focuses on the NEF functionality and proposes a novel, cloud-native approach for the Location Reporting feature based on the 3GPP standards, while at the same time minimizing dependencies on core network functions. The solution leverages an intelligent agent to monitor AMF registration logs, enabling the extraction and interpretation of location-related events without requiring direct interfaces with the core network itself. This design enhances flexibility and scalability and introduces a reduced dependency on direct 5G Core integration by using log-based extraction and external database storage of event information such as subscription management and asynchronous event notifications while at the same time adhering to modern cloud-native principles.

The remainder of the paper is organized in the following structure. Section II reviews the background concepts; Section III presents the proposed design and architecture of the API; Section IV details the implementation and the setup; Section V discusses the validation of the solution and finally, Section VI concludes the paper.

II. BACKGROUND ANALYSIS AND RELATED WORK

The concept of network exposure has undergone significant development across successive 3GPP releases. Initially introduced as the Service Capability Exposure Function (SCEF) in 4G networks, it evolved into the more robust NEF with the advent of 5G and its Service-Based Architecture (SBA). In the 5GC, the NEF acts as a vital bridge between internal network functions and external Application Functions (AFs), providing standardized APIs that allow secure access to network capabilities [6]. The services that are related to the UE lifecycle and the monitoring of an event shape one of the key interfaces, the Monitoring Event API, as specified in 3GPP TS 29.122 and TS 29.522, respectively [7], [8], which enables AFs to receive notifications about changes in User Equipment location (UE) location via the T8 reference point. This API supports a range of configurable parameters, such as the event type, the number of reports allowed, and the reporting interval—elements that form the core of the implementation described in this work.

Within the 5G network, and as far as the location is concerned, a UE undergoes different stages in terms of connectivity, availability, and location awareness. Sometimes, this kind of information may be valuable for an external 5G-enabled application. For example, an application needs to be aware of the location of its users within an industrial zone so as to make authorization decisions for the personnel. The monitoring of a UE's lifecycle by an external application within a 5GS (5G System) can be realized through the MonitoringEvent API. Some of the events under the scope of the aforementioned API include location reporting, UE reachability, and loss of connectivity [9].

Moving a step forward, 5G networks categorize the location services into three types based on which component initiates the location reporting. Network Induced Location Requests (NI-LR) are initiated by an AMF to determine a UE's location on behalf of a regulatory agency (e.g., to determine location during an IP Multimedia Subsystem (IMS) emergency call). Mobile Terminated Location Requests (MT-LR) are initiated by internal or external Location Service (LCS) clients, which can be third-party applications accessed by the UE, regulatory agencies, or mobile network operators. Last but not least, Mobile Originated Location Requests (MO-LR) are initiated by the UE to determine its own location or to provide its location to an external client.

The integration of cloud-native design principles in telecommunications networks has enabled the implementation of standardized interfaces like the NEF API. Technologies like containerization, microservice orchestration, and externalized state management have enabled the decoupling of network services and the development of more flexible, scalable architectures. The method proposed in the paper utilizes three fundamental cloud-native patterns to transform the delivery of NEF capabilities.

Given the principles of cloud-native design, such as scalability, flexibility, and loose coupling, certain architectural approaches are particularly well-suited for modern NEF imple-

mentations. Log-based monitoring, for instance, could reduce system coupling and improve observability by extracting UE-related events from AMF registration records, rather than relying on direct API interactions. External data persistence, achieved through dedicated databases like MongoDB, supports autonomous state management and enables stateless service components. Additionally, exposing interfaces through RESTful APIs defined using OpenAPI specifications promotes clarity, interoperability, and ease of integration across heterogeneous environments. Together, these features form a modular and flexible solution to the efficiency of NEF capabilities in next-generation networks.

To the best of our knowledge, NEF implementations, including those supporting the 3GPP-based monitoring event API, remain limited in number and scope. While prior work proposes a scalable, microservice-based NEF with Open Common API Framework (OpenAPIF) and demonstrated its interoperability with Open5GS, it does not implement Location Retrieval APIs [10]. Another work proposes a direct NEF-User Plane Function (UPF) control plane integration to optimize quality of service (QoS) provisioning and enable monetizable 5G services; however, this work does not touch the NEF Location Retrieval procedures [11]. The implementation presented in this paper addresses this gap by streamlining internal processes and simplifying interactions with the AMF for retrieving UE location information. In addition, the present research targets the location exposure domain, a sparsely investigated NEF API for location retrieval, which is crucial to mobility-aware and context-aware services in 5G/6G networks. This work offers a concrete design and testing approach, contributing a reproducible framework for additional research and open-source experimentation.

III. NEF MONITORING API DESIGN

In order for an AF to retrieve the location of a UE, appropriate event reporting parameters must be configured and a `Nnef_eventExposure_Subscribe` request to NEF must be issued. This request initiates a sequence of operations across various Network Functions (NFs) in the 5G Core, ultimately resulting in a location notification being sent from the NEF to the AF [12]. The high-level approach is illustrated in Figure 1. The baseline for our implementation builds upon prior work that introduced NEFSim, an open-source NEF simulator designed to enable testing of the northbound API in a simulated environment [13]. Building on this foundation, our work now extends NEFSim logic to support cloud-native deployment and real-time event handling and integration with AMF-derived data, utilizing the southbound interfaces. This section details the architecture and functional design of the proposed implementation.

A. Architecture Overview

The system is composed of two main modules: the AMF Log Agent (backend) and the NEF Monitoring Event API (frontend). Together, they implement a fully decoupled moni-

toring pipeline that remains transparent to the 5GC while still compliant with 3GPP TS 29.522 event monitoring [8].

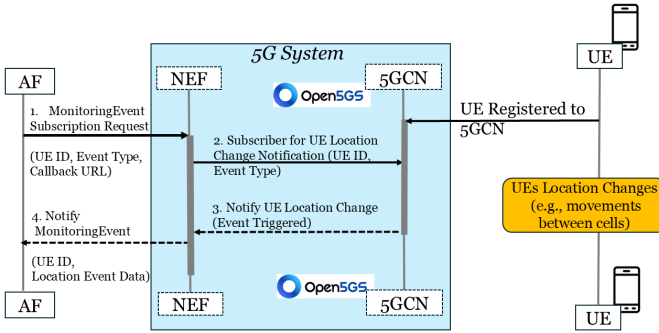


Fig. 1. High level overview of NEF Event Exposure in 5G Core.

1) *AMF Log Agent (Backend)*: The AMF Log Agent monitors UE registration activity by parsing AMF container logs. This agent employs a configurable polling strategy, customizable by the end-user, scanning for registration messages that contain relevant UE identifiers (e.g., IMSI). Upon detecting such an event, the agent parses the logs to extract relevant registration details:

- Raw registration metadata, including timestamps and event context.
- Parsed UE location information, formatted in alignment with 3GPP data models.

These data structures are constructed as JavaScript Object Notation (JSON) objects and stored in an external MongoDB database in two collections: `ue_events` and `location_info`. This separation enables persistent, queryable storage that is accessible by external NEF components.

2) *Monitoring Event Subscription and AMF Log Correlation*: The `MonitoringEventSubscription` message, as defined in 3GPP TS 29.122, enables an application server to request monitoring of specific network events for a target subscriber [7]. This subscription is characterized by several critical attributes: `MonitoringType`, which specifies the class of events to be monitored; `LocationType`, indicating the current or last known location of a UE, `msisdn`, uniquely identifying the subscriber of interest, `maximumNumberOfReports`, which constrains the total number of notifications generated for the subscription; `notificationDestination`, specifying the endpoint where monitoring notifications should be delivered; and `repPeriod`, which defines the periodicity for reporting events. Upon occurrence of a subscribed event, the network generates a `MonitoringNotification` message, which encapsulates one or more `MonitoringEventReport` instances. Each report provides detailed information regarding the monitored event and, if location spotted, includes a `LocationInfo` object that conveys the subscriber’s location according to the prescribed `LocationType`. This mechanism ensures that applications can obtain timely and structured information about network events

along with associated spatial context, enabling sophisticated analytics and context-aware services.

In parallel, log monitoring of AMF provides a mechanism to observe UE registration, deregistration, and PDU session establishment events within the 5G core network. By analyzing these logs, it is possible to extract network-derived attributes such as Tracking Area Identity (TAI), Cell ID, or other signaling context, thereby enabling coarse-grained location retrieval of the subscriber. This approach allows applications to correlate UE registration or session events with approximate location information, facilitating tracking of subscriber presence and network attachment patterns. However, the method is inherently limited to events that generate AMF-level signaling; Radio access network (RAN)-level mobility, may not trigger additional AMF logs, resulting in potential gaps in location granularity. To address this limitation, log monitoring can be complemented with RAN signaling events, AMF context updates, or dedicated 5G positioning mechanisms, providing higher-fidelity tracking while maintaining compatibility with Monitoring Event API subscriptions.

3) *NEF Monitoring Event API (Frontend)*: The frontend exposes an Asynchronous Representational State Transfer (RESTful) Monitoring Event API interface to AFs using OpenAPI specifications. It supports standard operations for subscription lifecycle management:

- `POST`: Creates a monitoring subscription with UE target ID and event parameters.
- `GET`: Retrieves subscription information by AF or subscription ID.
- `DELETE`: Cancels a monitoring subscription.

When a new `POST` subscription request is submitted, the NEF performs the following operations: First, it creates and stores an association between the subscription and its corresponding request parameters in a MongoDB database. This is the same database utilized by the AMF Log Agent to support other Hypertext Transfer Protocol (HTTP)-based operations. Subsequently, the NEF asynchronously queries the database for UE identifiers (IMSI) that match the subscription criteria. If a match is identified, the NEF retrieves the corresponding location information and delivers it to the AF via a predefined callback interface. This response is structured in accordance with 3GPP specifications [7], [8].

This decoupled architecture allows the NEF to provide location information without requiring real-time interaction with the 5G Core network. Such an approach simplifies system integration and enhances interoperability with different open-source 5G core network solutions.

B. Cloud-Native Deployment

The entire system is designed using state-of-the-art cloud-native technologies and principles. All components are containerized with Docker and orchestrated via Docker Compose. MongoDB serves as the central data store, with endpoints configurable at runtime. This design ensures platform independence, supporting integration with any compliant 5GC deployment without modification to the core network.

IV. API EXPOSURE VALIDATION

This section presents the practical deployment and integration of the NEF Monitoring Event API within a cloud-native 5G testbed. The implementation is based on open-source components and deployed entirely in a containerized environment.

A. Evaluation Setup

The complete testbed consists of the following elements:

- **5G Core:** We employ open5GS as the reference 3GPP-compliant 5G Core implementation. It provides the essential network functions, such as the Access and Mobility Management Function (AMF) and Session Management Function (SMF), enabling realistic simulation of UE registration and mobility procedures [14].
- **RAN Emulator:** The Radio Access Network (RAN) is emulated using UERANSIM, an open-source tool capable of simulating both User Equipment (UE) and gNodeB (gNB). It allows for the generation of realistic NAS signaling and interaction with the core network [15].
- **AMF Log Agent:** A custom-developed backend component designed to parse logs produced by the AMF. It extracts relevant registration and location data, which is then structured and stored for use by higher-layer services such as the NEF.
- **MongoDB:** A NoSQL database used to store registration metadata extracted from the AMF logs. This includes UE registration events, IMSI identifiers, and associated location information required by the NEF API.
- **NEF Monitoring Event API:** A RESTful API endpoint implemented to conform with 3GPP NEF specifications [1]. It supports monitoring event subscriptions and exposes UE location data through a documented Swagger interface for seamless AF integration.

Each component runs in isolated Docker containers, simulating a cloud-native 5G system. All components were developed and deployed using Docker. The environment configuration files enable full reproducibility, and the NEF API is exposed via the Swagger user interface (UI) for ease of interaction. This modularity supports easy redeployment in other cloud or on-premise 5G setups.

B. Operational Workflow

The system workflow unfolds in several stages, as figure 2 depicts. The first step begins with a simulated UE via the UERANSIM initiating a registration request to the Open5GS core [16]. As part of this process, registration logs are generated within the AMF container. Then, step 2 includes the AMF Log Agent, a custom backend component designed to continuously monitor AMF activity, which polls the AMF log files to identify registration events in near real-time. Upon detection, the agent extracts and parses critical data — including the IMSI (International Mobile Subscriber Identity), PLMN (Public Land Mobile Network) ID, and Cell ID — and formats the information into structured JSON objects. These objects are then persistently stored in a MongoDB instance,

specifically within collections dedicated to user events and location information.

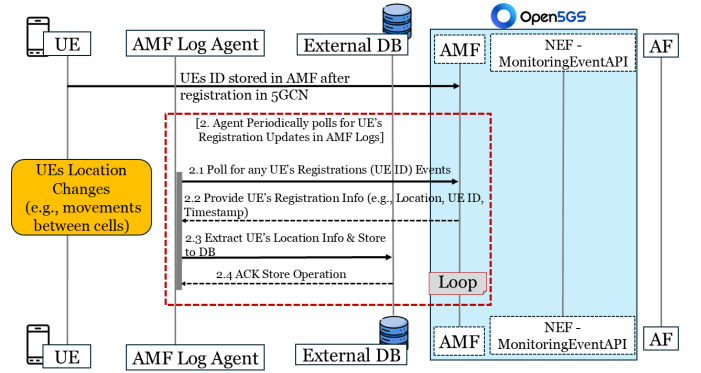


Fig. 2. Back-end Architecture for AMF Log Agent.

In addition, in step 3, in parallel, the external AF, integrated via a Swagger-based API interface, submits a POST request to the NEF Monitoring Event API. This request includes the target UE identifier along with subscription parameters such as event type and reporting configuration. The NEF API processes the request, validates the input, and stores the subscription metadata in the shared MongoDB database for future reference and matching against registered UEs. Furthermore, under step 4, once the subscription is stored, the NEF Monitoring API asynchronously checks for matching UE activity by querying the MongoDB collections populated by the AMF Log Agent. If a match is found between a stored subscription and a previously parsed UE registration event, the NEF constructs the location notification message. This message is then transmitted to the AF via a predefined HTTP callback interface.

Finally, in step 5, after receiving the notification, the AF is able to manage the subscription lifecycle through additional API operations. Specifically, the AF can retrieve subscription details using a GET request or cancel the subscription entirely with a DELETE request. Both operations are executed by the NEF and confirmed through appropriate API responses, thus finalizing the monitoring session.

V. RESULTS AND ANALYSIS

To validate the proposed NEF Monitoring Event API, we conducted a sequence of end-to-end tests based on the process described in the previous subsection. The objective was to confirm the correctness of subscription handling, log-driven event detection, asynchronous notification delivery, and full lifecycle management of monitoring subscriptions.

Figure 3 presents a snapshot of the backend log output captured during an experiment cycle. The sequence begins with the AF submitting a POST request to the NEF Monitoring API with a target IMSI identifier (001010143245445). This operation is acknowledged successfully, and the subscription parameters are stored in the shared MongoDB instance, as highlighted in the red box.

```

2025-07-04:09:54:50,858 INFO [cbs_data_handler.py:64] Document inserted successfully.
2025-07-04:09:54:50,859 INFO [monitoring_event_service.py:78] Response sent for AF POST request with IMSI 001010143245445
2025-07-04:09:54:50,860 INFO [reports_and_notification_helper.py:42] Processing report for IMSI: 001010143245445, report number: 1
INFO: 10.220.2.197:59789 - POST /3gpp-monitoring-event/v1/example/subscriptions HTTP/1.1 201 Created
2025-07-04:09:55:10,864 INFO [reports_and_notification_helper.py:43] Fetched Document: {'_id': '001010143245445', 'cellId': '000000010', 'trackingAreaId': '000001', 'plmnId': {'mcc': '001', 'mnc': '01'}, 'routingAreaId': None, 'endocBId': None, 'twanId': None, 'UELocationImestamp': '2025-07-04T09:47:37.842771Z'}
2025-07-04:09:55:10,867 INFO [reports_and_notification_helper.py:80] I have received the event report for subscription: http://10.220.2.73:8000/3gpp-monitoring-event/v1/example/subscriptions/cb23261c-f38d-4d0c-a795-0f5cbd042a96
2025-07-04:09:55:10,467 INFO [reports_and_notification_helper.py:85] Monitoring Notification: subscription=AnyHttpUrl('http://10.220.2.73:8000/3gpp-monitoring-event/v1/example/subscriptions/cb23261c-f38d-4d0c-a795-0f5cbd042a96') monitoringEventReport=MonitoringEventReport(externalId=None, msisdn='001010143245445', locationInfo=LocationInfo(ageOfLocationInfo=None, cellId='000000010', trackingAreaId='000001', endocBId=None, routingAreaId=None, plmnId=PlmnId(mcc='001', mnc='01'), twanId=None, geographicArea=None), locFailureCause=None, monitoringType='MonitoringType.LOCATION_REPORTING', eventIme=edatetime.datetime(2025, 7, 4, 9, 47, 37, 842771, tzinfo=TzInfo(UTC))) cancelInd=False
2025-07-04:09:55:10,511 INFO [reports_and_notification_helper.py:89] Response received successfully
2025-07-04:09:55:10,513 INFO [reports_and_notification_helper.py:12] Processing report for IMSI: 001010143245445, Report Number: 2
2025-07-04:09:55:30,517 INFO [reports_and_notification_helper.py:43] Fetched Document: {'_id': '001010143245445', 'cellId': '000000010', 'trackingAreaId': '000001', 'plmnId': {'mcc': '001', 'mnc': '01'}, 'routingAreaId': None, 'endocBId': None, 'twanId': None, 'UELocationImestamp': '2025-07-04T09:47:37.842771Z'}
2025-07-04:09:55:30,518 INFO [reports_and_notification_helper.py:80] I have received the event report for subscription: http://10.220.2.73:8000/3gpp-monitoring-event/v1/example/subscriptions/cb23261c-f38d-4d0c-a795-0f5cbd042a96
2025-07-04:09:55:30,569 INFO [reports_and_notification_helper.py:85] Monitoring Notification: subscription=AnyHttpUrl('http://10.220.2.73:8000/3gpp-monitoring-event/v1/example/subscriptions/cb23261c-f38d-4d0c-a795-0f5cbd042a96') monitoringEventReport=MonitoringEventReport(externalId=None, msisdn='001010143245445', locationInfo=LocationInfo(ageOfLocationInfo=None, cellId='000000010', trackingAreaId='000001', endocBId=None, routingAreaId=None, plmnId=PlmnId(mcc='001', mnc='01'), twanId=None, geographicArea=None), locFailureCause=None, monitoringType='MonitoringType.LOCATION_REPORTING', eventIme=edatetime.datetime(2025, 7, 4, 9, 47, 37, 842771, tzinfo=TzInfo(UTC))) cancelInd=False
2025-07-04:09:55:30,578 INFO [reports_and_notification_helper.py:89] Response received successfully
2025-07-04:09:55:30,576 INFO [reports_and_notification_helper.py:12] Processing report for IMSI: 001010143245445, Report Number: 3
2025-07-04:09:55:50,578 INFO [reports_and_notification_helper.py:43] Fetched Document: {'_id': '001010143245445', 'cellId': '000000010', 'trackingAreaId': '000001', 'plmnId': {'mcc': '001', 'mnc': '01'}, 'routingAreaId': None, 'endocBId': None, 'twanId': None, 'UELocationImestamp': '2025-07-04T09:47:37.842771Z'}
2025-07-04:09:55:50,578 INFO [reports_and_notification_helper.py:80] I have received the event report for subscription: http://10.220.2.73:8000/3gpp-monitoring-event/v1/example/subscriptions/cb23261c-f38d-4d0c-a795-0f5cbd042a96
2025-07-04:09:55:50,578 INFO [monitoring_event_service.py:29] This is the last notification for IMSI 001010143245445
2025-07-04:09:55:50,604 INFO [reports_and_notification_helper.py:85] Monitoring Notification: subscription=AnyHttpUrl('http://10.220.2.73:8000/3gpp-monitoring-event/v1/example/subscriptions/cb23261c-f38d-4d0c-a795-0f5cbd042a96') monitoringEventReport=MonitoringEventReport(externalId=None, msisdn='001010143245445', locationInfo=LocationInfo(ageOfLocationInfo=None, cellId='000000010', trackingAreaId='000001', endocBId=None, routingAreaId=None, plmnId=PlmnId(mcc='001', mnc='01'), twanId=None, geographicArea=None), locFailureCause=None, monitoringType='MonitoringType.LOCATION_REPORTING', eventIme=edatetime.datetime(2025, 7, 4, 9, 47, 37, 842771, tzinfo=TzInfo(UTC))) cancelInd=True
2025-07-04:09:55:50,660 INFO [reports_and_notification_helper.py:89] Response received successfully
2025-07-04:09:55:50,663 INFO [monitoring_event_service.py:36] Subscription cb23261c-f38d-4d0c-a795-0f5cbd042a96 deleted
2025-07-04:09:55:50,663 INFO [monitoring_event_service.py:30] Task registry: {}

```

Fig. 3. Location Exposure Status Responses

Subsequently, the AMF Log Agent detects a new UE registration event by parsing the AMF container logs. The extracted metadata includes the PLMN ID, cell ID, and the timestamp, which is transformed into a 3GPP-compliant JSON structure and persisted in the `location_info` collection. This process is shown in the orange-highlighted log entries. Upon matching the IMSI with the subscription entry, the NEF component triggers a location notification, invoking the callback interface provided by the AF.

Finally, as depicted in the yellow box, when the location notification event completes, the AF sends back a successful response, so the NEF backend initiates the automatic deletion of the subscription, completing its lifecycle.

VI. OPEN CHALLENGES

In open-source implementations of the 5G Core (5GC), extending the log-agent to systematically capture and monitor UE Registration events in alignment with 3GPP specifications enables interoperability and consistent monitoring of mobility and connectivity procedures across heterogeneous 5GC deployments. However, when a 5GC implementation diverges from strict 3GPP compliance through custom signaling flows, data structures, or non-standard logging formats, additional effort is required to map UE Registration events correctly. This involves analyzing source code and execution traces at the AMF level to extract relevant user location and registration context. Once established, the log-agent can normalize captured data into a standardized event representation, providing a unified, implementation-agnostic monitoring framework.

From a security and privacy perspective, the NEF must comply with 3GPP TS 33.501 to safeguard confidentiality and integrity of UE data, while CAPIF introduces additional requirements for secure and authorized API exposure [17]. Integration complexity in heterogeneous environments arises from varying levels of 3GPP compliance, requiring adaptation

logic to ensure interoperability without compromising security or privacy.

VII. CONCLUSIONS

This paper presented a novel, cloud-native implementation of the 3GPP NEF Monitoring Event API designed to enable scalable and decoupled UE location tracking for 5G and beyond. By leveraging AMF log parsing and external MongoDB storage, the proposed architecture minimizes direct interaction with the 5G Core, allowing for flexible deployment across heterogeneous environments without sacrificing 3GPP compliance.

The system supports full subscription lifecycle management through a RESTful API interface and achieves real-time, asynchronous delivery of UE location notifications to third-party application functions. This open-source solution lowers the entry barrier for research institutions and smaller network operators looking to experiment with NEF-based services. It also provides a future-proof framework adaptable to the evolving needs of 6G networks, where modularity, vendor neutrality, and high observability will be critical enablers for intelligent, location-aware applications.

Future work will focus on extending support to additional 3GPP monitoring event types, integrating edge intelligence for proactive analytics, and evaluating quantitative performance metrics such as end-to-end latency, signaling load, and API throughput under varying subscription densities. Furthermore, extending the NEF MonitoringEvent API within CAPIF-secured environments will provide a standardized and robust foundation for event monitoring across diverse 5GC implementations, while the modular, containerized design ensures seamless integration and scalability in edge and cloud deployments.

ACKNOWLEDGMENT

The work has been partially funded by the Smart Networks and Services Joint Undertaking (SNS JU) under the Euro-

pean Union’s Horizon Europe research and innovation programme for the ENVELOPE, SAFE-6G, UNITY-6G, and 6G-VERSUS project’s under Grant Agreement’s No.101139048, No.101139031, No.101192650, and No.101192633 respectively.

REFERENCES

- [1] G. Makropoulos, D. Fragkos, H. Koumaras, J. Cijan, L. Koršič, and R. Sušnik, “5g and b5g nef exposure capabilities towards an industrial iot use case,” in *2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, 2023, pp. 647–651.
- [2] D. Tsolkas and H. Koumaras, “On the development and provisioning of vertical applications in the beyond 5g era,” *IEEE Networking Letters*, vol. 4, no. 1, pp. 43–47, 2022.
- [3] L. Bonati, M. Polese, S. D’Oro, S. Basagni, and T. Melodia, “Open, programmable, and virtualized 5g networks: State-of-the-art and the road ahead,” *Computer Networks*, vol. 182, p. 107516, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1389128620311786>
- [4] S. Eswaran and P. Honnavalli, “Private 5g networks: a survey on enabling technologies, deployment models, use cases and research directions,” *Telecommunication Systems*, vol. 82, no. 1, pp. 3–26, 2023. [Online]. Available: <https://doi.org/10.1007/s11235-022-00978-z>
- [5] G. Makropoulos, D. Fragkos, H. Koumaras, N. Alonistioti, A. Kaloxylas, V. Koumaras, T. Dounia, C. Sakkas, and D. Tsolkas, “5g network programmability enabling industry 4.0 transformation,” in *Opportunities and Challenges of Industrial IoT in 5G and 6G Networks*, P. Yu, X. Hu, A. Prakash, N. Misuko, and G. Haiyue, Eds. Hershey, PA: IGI Global Scientific Publishing, 2023, pp. 119–137. [Online]. Available: <https://doi.org/10.4018/978-1-7998-9266-3.ch006>
- [6] F. Setaki, I. Mesogiti, E. Theodoropoulou, G. Lyberopoulos, H. Koumaras, D. A. Guillen, J. G. Rodrigo, G. Avdikos, I. Margaritis, E. Kafetzakis, Y. Karadimas, and D. Tsolkas, “The netapps certification environment for 5g and beyond vertical ecosystems: The evolved-5g approach,” in *2022 IEEE Globecom Workshops (GC Wkshps)*, 2022, pp. 1182–1187.
- [7] 3GPP, “TS 29.122: Service Capability Exposure Function (SCEF); Stage 3,” 3rd Generation Partnership Project (3GPP), Technical Specification TS 29.122, 2023.
- [8] 3GPP, “TS 29.522: Common API Framework for 3GPP Northbound APIs,” 3rd Generation Partnership Project (3GPP), Technical Specification TS 29.522, 2023.
- [9] D. Fragkos, G. Makropoulos, P. Sarantos, H. Koumaras, A.-S. Charismiadis, and D. Tsolkas, “5g vertical application enablers implementation challenges and perspectives,” in *2021 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*, 2021, pp. 117–122.
- [10] J. Fernandes, L. Rosa, J. Gameiro, P. Valente, E. Ramos, J. Oliveira, D. Raposo, A. T. Ferrer, J. M. Portela, P. Rito, L. Cordeiro, and S. Sargento, “Pushing the boundaries of scalable 5g core networks: Cloud-native nef and capif interplay,” in *2024 3rd International Conference on 6G Networking (6GNet)*, 2024, pp. 201–205.
- [11] A. S. L and M. Patwary, “Optimizing 3gpp 5g nef integration: Enhancing programmable networks through nef and upf interoperability,” in *2024 IEEE Future Networks World Forum (FNWF)*, 2024, pp. 423–427.
- [12] 3GPP, “TS 23.502: Procedures for the 5G System (5GS),” 3rd Generation Partnership Project (3GPP), Technical Specification TS 23.502, 2023.
- [13] D. Fragkos, G. Makropoulos, A. Gogos, H. Koumaras, and A. Kaloxylas, “Nefsim: An open experimentation framework utilizing 3gpp’s exposure services,” in *2022 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, 2022, pp. 303–308.
- [14] E. Aumayr, G. Caso, A.-M. Bosneag, A. D. Zayas, Özgü Alay, B. Garcia, K. Kousias, A. Brünstrom, P. M. Gomez, and H. Koumaras, “Service-based analytics for 5g open experimentation platforms,” *Computer Networks*, vol. 205, p. 108740, 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1389128621005892>
- [15] U. Team, “UERANSIM: Open-source 5G UE and RAN simulator,” <https://github.com/algungr/UERANSIM>, 2023, [Online; accessed 15-Aug-2023].
- [16] H. Koumaras, D. Fragkos, V. Pitsilis, G. Makropoulos, G. Lyberopoulos, E. Theodoropoulou, F. Setaki, I. Mesogiti, D. Diakakis, N. Gkatzios, C. Fragkos, and V. Koumaras, “The b5g athens experimentation platform,” in *2024 IEEE 29th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2024, pp. 1–2.
- [17] 3GPP, “TS 33.501: Security architecture and procedures for the 5G System (5GS),” 3rd Generation Partnership Project (3GPP), Technical Specification TS 33.501, 2024.