ISSN-L 2664-9969; E-ISSN 2706-5448

UDC 004.75:004.056:614.21]:519.85
DOI: 10.15587/2706-5448.2026.358187

Ivan Shevtsov,
Tetiana Fesenko

DEVELOPING A TASK ALLOCATION MODEL FOR REMOTE HEALTH MONITORING IN SMART CITIES CONSIDERING LATENCY, ENERGY CONSUMPTION, AND PRIVACY ON FOG NODES

The object of the research is the processes of dynamic distribution of computing tasks in multi-level infrastructures of a smart city. The possibilities of integrating edge, fog, and cloud computing resources for the development of remote patient monitoring systems (Remote Patient Monitoring, RPM) were investigated. The study addresses the challenge of balancing the rapid processing of critical medical signals with the limited energy resources of mobile devices. In addition, the need to ensure the confidentiality of personal data when transferring tasks to third-party fog nodes was addressed through encryption, remote attestation mechanisms, and isolated execution environments.

A comprehensive system model was developed to describe the processes of performing RPM tasks (ECG classification, audio analysis). An offloading strategy was developed, based on a weighted linear to minimize energy consumption and delay. An architectural framework is proposed to ensure the confidentiality of data processing on uncontrolled fog nodes, through the use of Trusted Execution Environment (TEE) technologies and the deployment of Trusted Applications (TA). To validate the solutions, a series of simulations was conducted in the YAFS (Yet Another Fog Simulator) environment to compare Mobile, Hybrid, and Fog scenarios.

It was experimentally established that transitioning to a Fog-oriented strategy results in a radical reduction in the average system latency (from 0.57 s to 0.027–0.030 s). The load on the smartphone is reduced by more than 10 times (from 222–225 mWh to 20.3–20.4 mWh), and the autonomy of wearable sensors increases almost fivefold. It is proven that the use of fog computing provides stable Quality of Service (QoS) on equipment with lower power (500 MIPS). The integration of attestation procedures according to RATS (Remote ATtestation procedureS) standard is intended to enable verification of the integrity of the computing stack before the transfer of confidential data.

Keywords: *smart city, fog computing, remote health monitoring, energy efficiency, confidentiality, data distribution.*

Received: 26.01.2026
Received in revised form: 24.03.2026
Accepted: 18.04.2026
Published: 30.04.2026

How to cite
Shevtsov, I., Fesenko, T. (2026). Developing a task allocation model for remote health monitoring in smart cities, considering latency, energy consumption, and privacy on Fog nodes. Technology Audit and Production Reserves, 2 (2 (88)), 34–47. https://doi.org/10.15587/2706-5448.2026.358187

1. Introduction

Analysis of research on remote patient monitoring systems (RPM) demonstrates a paradigm shift from stationary-oriented solutions to mobile ecosystems of continuous monitoring. During the COVID-19 pandemic and limited physical access to clinics/hospitals/healthcare facilities, the digitalization of medical services has gained rapid development [1]. Scientific and practical developments suggest combining the capabilities of wearable gadgets, cloud computing, and intelligent networks to analyze medical indicators in real time. The paper [2] proposes the creation of mobile ecosystems where "clouds", digital gadgets (which patients wear to read data), and smart networks work as a single whole. This, in turn, allows for the analysis of biometric indicators/data in real time. Such architecture provides incredible flexibility, but at the same time, overloads communication channels. To reduce the load on device batteries, resource-intensive operations can be transferred to the network edge, to the edge and fog levels.

The implementation of RPM services for the long-term management of chronic patients currently relies on the Internet of Things (IoT) infrastructure [3]. In the modern e-health ecosystem, these technologies provide the collection and transmission of the necessary data. The Singaporean program HealthTrack SG demonstrates that Smart Health's success depends on the system's ability to aggregate clinical data from portable devices without interruption [4].

An important direction in the development of intelligent healthcare systems is the integration of electronic health records (EHR) with streaming data from IoT sensors [5]. This allows the system not only to collect data, but also to automatically generate personalized clinical advice/recommendations. In addition to running complex diagnostic models, there is a need for dynamic task allocation (Task Allocation). Resource-intensive data analysis should be implemented on a fog infrastructure so that the patient's medical profile remains confidential.

Thus, digital medicine in a smart city is a complex hierarchy of edge devices, fog nodes, and cloud services [6]. A distributed approach helps the system scale and withstand strict latency requirements. Fog Computing brings analytics closer to the data source and offloads backbone networks. In this regard, the issues of optimization and balancing among minimizing delays in the transmission of critical biosignals, minimizing energy consumption of mobile terminals, and ensuring data confidentiality are becoming increasingly relevant.

In Ukraine, the use of telemedicine technologies is regulated by the "Procedure for the Organization of Medical Care at the Primary, Secondary (Specialized), and Tertiary (Highly Specialized) Levels Using Telemedicine" [7]. This document defines the requirements for the telemedicine network and its tools (telemedical consultation, telemedical consultation journal, telemedical council, telemetry, home teleconsultation). Particular attention is paid to maintaining the confidentiality and integrity of medical information, as well as the requirements of ethics and deontology of medical care. Telemedicine resource support includes specialized software and hardware to implement the functions of the Telemedicine Portal, video and audio communication, telemetry tools, etc. The priorities in the state policy of e-Health reform in Ukraine are digitalization and expansion of electronic medical services. The development of dynamic task distribution models [8] allows adapting national medicine to the requirements of Big Data systems, where every millisecond of delay in monitoring an acute condition can become critical. The scaling of digital healthcare ecosystems within the framework of the smart city concept is accompanied by an exponential increase in the volume of streaming information. This forces more stringent requirements for the speed of response when monitoring acute conditions. The traditional model of transmitting raw biosignals from wearable sensors directly to remote cloud data centers reveals a number of destructive factors: a significant increase in latency, high sensitivity to jitter in wireless communication channels, and premature depletion of the energy resource of mobile terminals. These limitations actualize the transition to hierarchical data processing models, where the intermediate level of fog computing allows to ensure a balance between the speed of system response and the autonomy of client devices, while creating the prerequisites for local verification of data confidentiality.

To overcome the technological limitations of centralized cloud systems, the OpenFog architectural concept proposes the implementation of a multi-level computing hierarchy, which involves delegating time-critical and resource-intensive operations from the cloud infrastructure to the edge of the network and intermediate fog nodes [9]. According to the OpenFog reference architecture, such a distribution provides the necessary local computing power for data processing near the sources of their generation. This allows not only to minimize network delays, but also to guarantee the continuity of medical services (reliability) and implement mechanisms for secure task execution in a heterogeneous environment, which is critically important for remote monitoring systems in the infrastructure of a smart city.

One of the most promising approaches to designing modern IoT systems is the implementation of multi-layer fog computing architectures (Multi-Layer Fog Computing, MLFC). The main idea of MLFC is to differentiate data processing depending on the complexity of the calculations. The study [10] shows how the use of several hierarchical levels of fog nodes adds the necessary flexibility to the system. Such a solution is promising for urban health monitoring services. It allows for parallel processing of both simple telemetric indicators and resource-intensive biosignals. However, the real operation of such models still does not solve the problem of energy shortage on patients' smartphones, and also does not guarantee sufficient isolation of processes to protect privacy.

The development of energy-efficient medical systems in foggy environments often relies on dynamic module placement strategies. These modules aim to minimize the total energy consumption of the entire infrastructure without sacrificing Quality of Service (QoS). In [11], it was proven that competent optimization of the distribution of components between intelligent nodes can significantly extend the battery life of gadgets and relieve unnecessary load on smart city networks. In hybrid "Cloud/Fog of Things" (CoT) ecosystems, the key challenge for RPM services remains the balance between energy consumption and latency. A systematic review of CoT solutions [12] confirms that the life of sensors and the speed of response to changes in patient biosignals are the main criteria/indicators of QoS quality. Analysis of offloading strategies in Multi-access Edge Computing (MEC) architectures demonstrates the priority of energy saving. The authors of [13] note that typical MEC models often ignore the specifics of RPM systems. In particular, the direct connection between energy saving and guaranteed confidentiality is ignored. In [14] also points out gaps in the study of energy profiles of end devices (smartphones and watches). In RPM scenarios, the frequency of access to fog nodes directly depends on the patient's health status, which makes the load stochastic. Positive results in optimizing infrastructure energy consumption are demonstrated by the use of metaheuristics, in particular the Firefly Algorithm [15]. However, such models are mainly focused on abstract types of loads and do not take into account the strict time constraints (deadlines) characteristic of the transmission of critical medical data. Similar problems are also manifested when using the LPDC (Latency-optimal and Priority-aware Deadline-Constrained) approach [16]. It provides efficient allocation of computing resources in heterogeneous networks taking into account dynamic node movement. However, the issues of terminal power consumption decomposition and biometric protection during offloading remain open. Predictive methods are effective in overcoming the uncertainty of the network context. For example, the PORA (Predictive Offloading and Resource Allocation) algorithm [17] is effective in overcoming network uncertainty by controlling the stability of queues. Instead, the algorithm focuses mainly on the stability of infrastructure nodes, and not on the energy resources of portable sensors. Also, progressive NOMA (Non-Orthogonal Multiple Access) technologies in combination with Energy Harvesting [18] are still based on simplified load profiles. That is why there are difficulties in their application in real scenarios of digital medicine of a smart city due to the lack of mechanisms to guarantee data integrity.

A comprehensive analysis of the technical barriers to RPM, conducted in [19], identifies measurement errors and privacy risks as the main obstacles. Studies [20, 21] prove the advantages of hybrid Edge/Fog models for reducing pressure on the cloud. At the same time, a mathematical description of the optimization that would take into account the energy of mobile devices and mechanisms for remote node attestation has not yet been presented. This creates a scientific basis for the development of an improved model that would integrate privacy parameters directly into the calculation scheduling algorithms. To ensure privacy at the network edge, it is critical to implement standardized interfaces for trusted execution environments (TEE). According to the GlobalPlatform TEE specification [22], the unification of system calls allows to guarantee the portability of secure modules between different hardware platforms. In the architecture of a smart city, this makes it possible to implement dynamic offloading of sensitive data into trusted containers. The methodological basis was the RATS remote attestation procedure (RFC 9334) [23], which allows to verify the state of a remote node before entrusting it with biometric processing. The use of RATS conceptual roles in RPM systems (in particular, the *Verifier* and the *Relying Party*) allows to form an objective evidence base to confirm the security of the fog infrastructure. A detailed analysis of attestation mechanisms for TEE [24] helps to classify these approaches and identify weaknesses in existing protocols. The authors of [24] consider in detail the limitations of TEE hardware implementations, which is fundamentally important when designing a heterogeneous smart city infrastructure, where fog nodes can have different levels of security. Taking these threats into account in the proposed task distribution model allows integrating the attestation stage as a necessary condition before offloading medical data. This ensures not only the confidentiality of processing in isolated enclaves but also the resistance of the RPM system to man-in-the-middle attacks and compromise of peripheral equipment, which is often ignored in classical models of optimization of latency and energy consumption.

Thus, a relevant scientific task is to develop a model of RPM task placement, which should simultaneously take into account the latency

ISSN-L 2664-9969; E-ISSN 2706-5448

indicators, energy consumption of mobile devices, and the possibility of confidential execution at the fog level.

The object of research is the processes of dynamic distribution of computational tasks in multi-tier infrastructures of a smart city.

The aim of research is to develop and substantiate a model of RPM task distribution in a multi-tier infrastructure of a smart city (Edge-Fog-Cloud), which provides a manageable compromise between latency and energy consumption of mobile devices through a composite criterion. The model defines a class of online offloading tasks for further optimization or machine learning methods, with validation on simulations of three architectural scenarios (mobile, hybrid, fog) and analysis of static placement policies.

Achieving the aim involves performing the following objectives:

1) to propose a system model for distributing RPM tasks in a smart city, taking into account task classes, deadlines, network/resource constraints, and a composite criterion (latency and power consumption of mobile devices);

2) to develop a framework for confidential execution on fog nodes based on TEE, remote attestation, Trusted Application, and least privilege policies;

3) to perform simulation experiments in YAFS (Yet Another Fog Simulator) with an extended power consumption model and Wi-Fi channel to compare scenarios and confirm the adequacy of the model. Test three architectural scenarios (mobile, hybrid, fog) that reflect the evolution from cloud-centric to edge-fog model.

2. Materials and Methods

2.1. Research methods

A comprehensive scientific approach was used to develop a model that provides a compromise between latency, energy consumption, and confidentiality. At the initial stage, the architectural features of remote patient monitoring (RPM) systems were investigated using the system analysis method. This allowed us to identify critical functional limitations of existing offloading models, which often do not take into account the specifics of medical traffic and high security requirements in the dynamic environment of a smart city.

Further formalization of resource constraints of nodes and description of the stochastic nature of medical traffic generation, in particular ECG signals and audio data, was implemented using mathematical modeling. Such an approach is necessary to transform a complex task allocation process into a multi-criteria optimization problem. The use of weighting coefficients within the model allows for flexible adjustment of the priority between the system response speed and the autonomy of patient devices. When developing the optimization criterion itself, decomposition and composition methods were used. They made it possible to separate the integral indicators of latency and energy consumption into separate components, with their subsequent combination into a linear weighted indicator. This is critically important for making informed decisions in a heterogeneous environment.

The construction of the architectural framework was based on the structural-functional design method using Trusted Execution Environment (TEE) technologies. The choice of this method was due to the need to ensure hardware isolation of calculations on public fog nodes to protect the privacy of medical data. Within this stage, security system design methods were used to implement GlobalPlatform TEE standards and RATS remote verification specifications. This allowed implementing equipment integrity verification procedures and coordinating technical solutions in accordance with the regulatory framework, in particular, the requirements for the protection of medical information.

Experimental validation of the proposed scenarios (mobile, hybrid, fog) was carried out by simulation modeling in the YAFS environment. Since the deployment of a real urban fog computing infrastructure is costly, the use of a simulator with an extended Wi-Fi channel model allowed for adequate reproduction of the network dynamics and assess the scalability of the model. The final stage was the comparative analysis method, which was used to verify the obtained results. By directly comparing the developed solution with traditional cloud-centric approaches, the effectiveness of the model was quantitatively confirmed, in particular in terms of reducing data transmission latency.

2.2. System input data and software

A set of input data was used to quantitatively evaluate the proposed model, including five medical tasks (*T1–T5*): ECG classification (*T1*), audio analysis (*T2*), temperature measurement (*T3*), humidity (*T4*), and ECG signal compression (*T5*). For each task, the input and output data volume, computational complexity in millions of instructions (MI), and the arrival period are specified. The corresponding parameters used in the simulation experiments are summarized in Table 1.

Table 1

Parameters of RPM computational tasks in e-health scenarios

ID	Load type (scenario)	Period, s	Complexity, MI	Input, KB	Deadline, s	Result volume, B
T1	ECG classification	10	500	36	15	740
T2	Audio analysis	10	1000	500	–	4096
T3	Temperature	60	10	10	0.5	707
T4	Humidity	60	10	10	0.5	707
T5	ECG compression	10	211	240	0.5	17203

In the simulation environment, the hardware configuration of the system includes a Samsung Galaxy S4 smartphone, an LG Urbane Watch smartwatch, and a fog node based on the PCEngines ALIX 3D2 platform. For each device, the processor performance (MIPS), CPU power, and power consumption profiles of Wi-Fi/BLE wireless interfaces obtained from empirical studies were taken into account. The software part of the research was implemented using the YAFS discrete-event simulator version 3.0. The simulator developed a multi-level Cloud-Fog-Edge architecture, traffic generation modules for tasks *T1–T5*, and also integrated advanced Wi-Fi channel and power consumption models.

The network configuration covers three architectural scenarios: *mobile* (performing tasks on the smartphone), *hybrid* (distributing processing between Edge and Fog), and *fog* (performing the main tasks on the fog node). The Wi-Fi channel modeling is based on fixed RSSI levels (–40 dBm, –70 dBm, –85 dBm) with linear approximation of intermediate values of throughput, TCP retransmission fraction, and power profiles. The calculations were performed on a system with an Intel Core i7-7920HQ processor (3.10 GHz), 16 GB RAM, running macOS Ventura.

The research is based on a number of assumptions and has limitations. First, the model does not take into account the procedure for migrating task state between nodes and the cost of establishing connections, which limits the analysis to stationary scenarios. Second, the impact of security mechanisms (attestation, TEE initialization, SELinux policies) on latency is not quantitatively modeled. It is possible to assume that the selected nodes have already passed verification. Third, medical traffic sources are described by deterministic periodic profiles, which does not capture the full clinical variability of patient behavior.

3. Results and Discussion

3.1. RPM task distribution system model

The system under study is based on a multi-layered Cloud-Fog-Edge-IoT infrastructure that integrates biometric sensors, mobile user terminals, peripheral fog nodes (Fog nodes), and cloud services (Fig. 1).

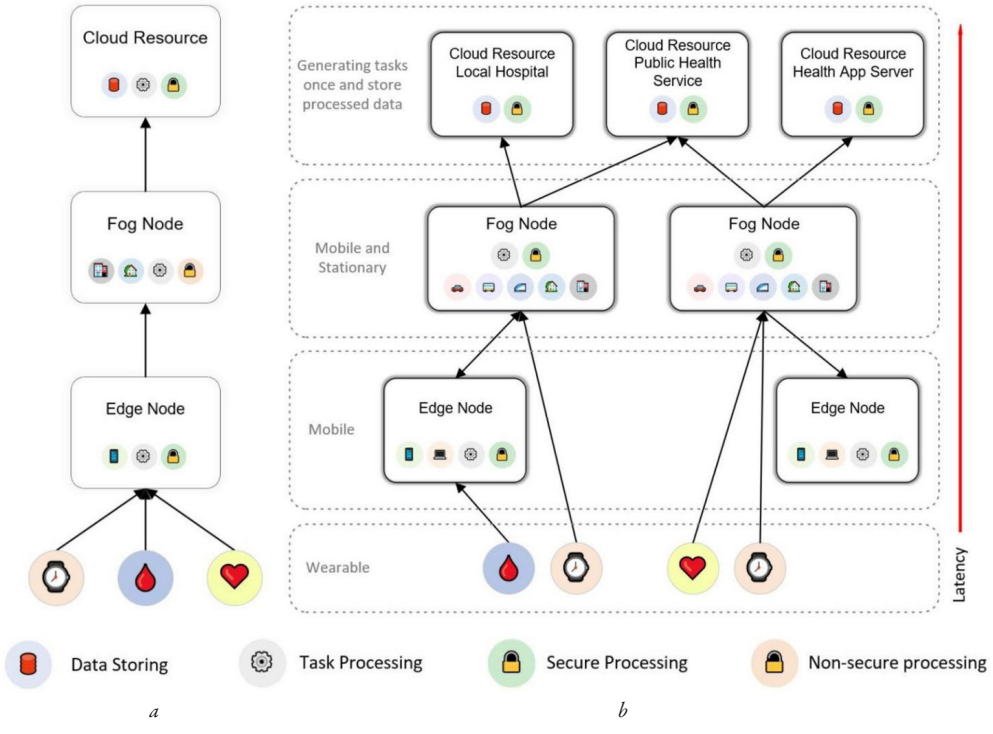

Fig. 1. Schematic representation of remote health monitoring models: *a* – existing model (state "as is"); *b* – proposed model (state "as should be")

In classical hierarchies, devices (which patients can wear) function as sources of periodic data that are directed to the edge node (smartphone) for primary processing. The shortage of local resources delegates/transfers the task to the fog layer, and in case of further lack of capacity – to the cloud.

Such a cascaded transfer causes a critical increase in latency, which is unacceptable for real-time RPM systems.

In addition, traditional fog nodes (home or hospital routers) are considered as undercooked environments, which limits their use for processing confidential medical data.

Unlike existing solutions, in the proposed model, the cloud layer (medical institutions, government services) acts as a generator of static computing tasks. These tasks (analysis algorithms) are deployed once on the edge node and updated only when the processing logic needs to be modified. This minimizes constant traffic between the Edge and the Cloud.

Key features of the proposed model are:

1) *connectivity flexibility:* the model provides for the possibility of direct connection of sensors both to the edge node and directly to the fog node via broadband protocols (Wi-Fi, 4G/5G). This allows optimizing the power consumption of the mobile device and reducing the overall latency in the network;

2) *rejection of cloud offloading:* transferring computational tasks back to the cloud is not considered, since the concept of the model is aimed at localizing processing as close as possible to the data source (Data Locality) to ensure deterministic response time;

3) *confidentiality through TEE:* a trusted execution environment (Trusted Execution Environment, TEE) is integrated for processing sensitive medical data. This allows the safe use of public infrastructure (city routers, on-board transport computers) as fog nodes;

4) *two-stage verification:* the security of the environment is guaranteed by the Remote Attestation mechanism. The process includes checking the integrity of the node hardware platform before deployment and further verification of the software application itself (Trusted Application) for the absence of compromise.

Fig. 2 shows the detailed architecture of the proposed framework, illustrating the interaction of levels and data flows for confidential execution of RPM tasks. The proposed model has a four-level hierarchical structure in which each level performs specific functions of processing and ensuring information protection.

The first – Perception Layer consists of a heterogeneous network of wearable and stationary medical devices (ECG sensors, smart watches, environmental monitoring sensors). Data is generated in the form of periodic or event-driven signals and transmitted to the next layer via BLE, ZigBee, Wi-Fi interfaces.

The second – Edge Layer is represented by the user's mobile terminal (smartphone), which functions as an intelligent gateway. This layer implements the storage of computational tasks received from the fourth (cloud) layer. Tasks stored at this layer are directly related to the user. In addition to the fact that the edge node has the ability to perform tasks locally, it also acts as an initiator of offloading, making decisions about delegating calculations to the Fog layer based on the current state of resources and latency requirements.

The third – Fog Layer forms a distributed network of nodes with limited, but significant resources (home gateways, city routers, cloudlets-type microservers). The main feature of this layer is the presence of a Trusted Execution Environment (TEE), which allows isolating task processing from the edge node from a potentially compromised operating system. This ensures the confidentiality of calculations even on public infrastructure.

The fourth – Cloud Layer is formed from high-performance server clusters designed for long-term storage of depersonalized big data (Big Data), global predictive analytics, and training of machine learning models [6, 10, 25, 26]. In the proposed model, cloud resources are not involved in the operational decision-making cycle, which minimizes network jitter.

The architecture of the model involves the use of secure transmission channels and a mandatory remote attestation procedure before each offloading session. This guarantees that confidential medical algorithms are executed only on those Fog nodes which integrity is confirmed at the hardware level.

ISSN-L 2664-9969; E-ISSN 2706-5448

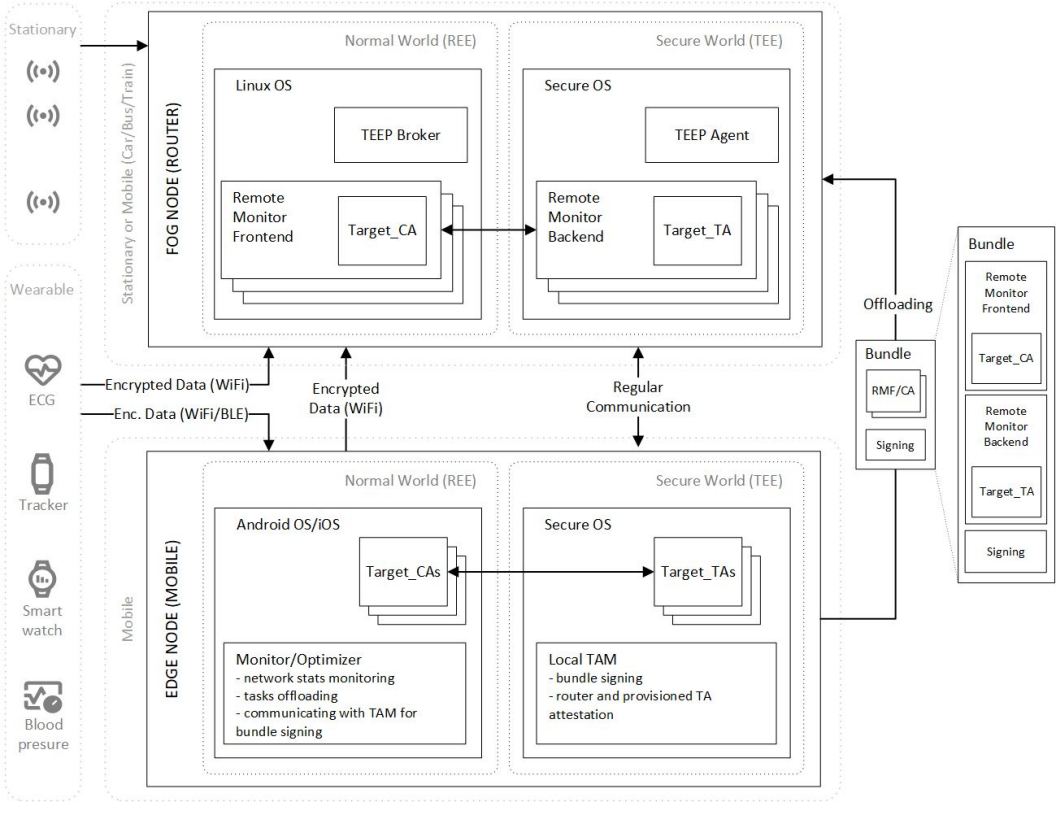

Fig. 2. Architecture of the framework for confidential execution of remote health monitoring tasks on edge, fog nodes, and data streams

For the mathematical description of the model, the following notations were used:

K – set of RPM task classes (ECG classification, audio analysis, contextual tasks, etc.); for each class $k \in K$, the data volume L_k is given (for example, the size of an ECG segment, audio fragment, etc.);

T_k – maximum response time, reflecting the latency requirements for the corresponding medical service (critical, near-real-time, non-critical);

T_k – generation period, determining the frequency of new data from sensors and, accordingly, the task execution period (it is possible to assume that the calculation task execution period is equal to the data acquisition period from the corresponding sensor);

π_k – priority, can be used to differentiate processing in conditions of limited resources.

If necessary, the task class can be tied to a certain quality of service class (QCI) in the network [27, 28].

The task flow of each class is modeled as a sequence of requests t_i. Each of them must be placed on one of the potentially available nodes $PN \in PNs_{all}$ (edge or fog nodes, cloud) at discrete times Δt_i.

The network is described by a directed graph $G = (N, E)$, where N – nodes (sensors, edge, fog, cloud), and E – transmission channels (BLE, Wi-Fi, cellular connections). For each task t_i, a set of admissible configurations $X(t_i)$ is considered, each of which corresponds to a route $p(X) \in E$ and the choice of a processing node.

The task execution time in configuration X is given as

$$T(X) = T_{tx}(X) + T_{CPU}(X), \quad (1)$$

where T_{tx} includes the components of transmission delays (transmission, processing, queues, retransmissions); T_{CPU} – the computation time on the selected node, which depends on the task size and the available performance (MIPS) of this node.

The energy consumption of a mobile device is modeled as

$$E(X) = E_{link}(X) + E_{CPU}(X), \quad (2)$$

where E_{link} – the integral of the transmit/receive power of Wi-Fi/BLE interfaces, which depends on RSSI (according to empirical profiles) [29]; E_{CPU} – the energy of local computations in the case of edge-level task execution. For wearable sensors, the energy of radio interfaces is similarly taken into account; however, the central role in the quality criterion is played by the energy consumption of the smartphone.

The online formulation of the problem is that for each task t_i at time Δt_i it is necessary to choose a configuration $X \in X(t_i)$ that minimizes the weighted trade-off criterion between the energy consumption of mobile devices and the delay in the task execution time

$$J(X) = \alpha E(X) + \beta T(X), \quad (3)$$

subject to the following constraints:

– each task is assigned to only one node from the admissible set $PNs(t_i)$;

– CPU utilization and memory usage on each node do not exceed the permissible limits;

– for tasks with deadlines, $T(X) \le d(t_i)$ is satisfied.

3.2. Framework for confidential task execution on fog nodes

For the use of fog nodes in remote health monitoring, not only the efficiency of task placement is critical, but also the confidentiality and integrity of medical data processing on potentially partially untrusted devices. A framework for confidential task execution on fog nodes is proposed, which combines the mechanisms of a trusted execution environment (TEE) [22, 24], hardware/software platform attestation, and the principle of least privilege for client applications. A fog node is considered as a node that can perform calculations only after verifying its state and the state of a trusted application, while the user's mobile device acts as a broker and local verifier. Fig. 3 shows a secure world configuration on Armv8-A using Secure Partition Manager (S-EL2) [30] and separating Trusted OS and specialized Secure Partitions, which corresponds to the recommended FF-A profile for maximum secure execution.

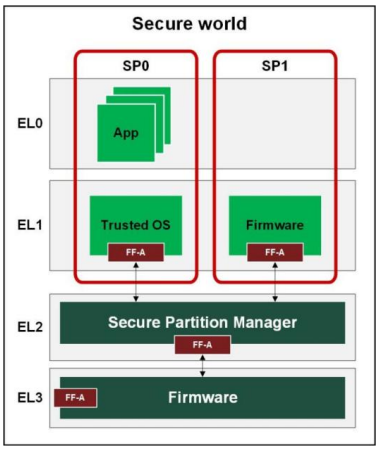

Fig. 3. Configuration of application placement in a protected world, which guarantees the greatest possible physical and logical isolation in the Armv8-A architecture

The proposed architecture distinguishes two main components: a client application (Client Application, CA), which runs in the regular execution environment (REE) of the router or gateway, and a trusted application (Trusted Application, TA), which runs inside the TEE. The mobile device receives the target TA and the associated CA from the developer, forms an installation package (bundle) in its own TEE. which includes the monitor and Target_TA. After that, the mobile device creates a manifest with access policies (Role-ID) and signs the package with the key of a trusted party, for example, a municipal operator or device owner. The received update package is sent to the fog node using a TEEP-type protocol, while the mobile device controls both the session initialization process and the subsequent state of the node [31]. The sequence of basic steps for deploying and attesting a CA/TA on a router is shown in Fig. 4.

Before deploying Target_TA on a fog node, remote platform attestation is performed [32]. At the challenge stage, the mobile broker sends the node a request with a random value (nonce) and a requirement to provide attestation evidence. The process of remote router attestation and involving a trusted party (TAM) and a local verifier on a mobile device is illustrated in Fig. 5. The fog node runs on a common operating system (e. g., embedded Linux) with a built-in hardware trust module (TPM or similar). In response, it generates a message containing platform register status indicators (PCR), a boot event log, and a signature calculated with a secure key. Next, the TEE on the router generates a separate report on the Target_TA, which records its code hash, version, and one-time public key for further secure exchange. The mobile device, having received the combined evidence package, checks it against the reference values ("golden image"), checks the freshness of the nonce, and the correctness of the signatures. Only if all checks are successful, the fog node and the specific Target_TA are considered trusted to perform the tasks.

After positive attestation, the components are deployed, and isolation is strengthened. A detailed scenario for preparing and installing Target_TA using the TEEP protocol is shown in Fig. 6. On the fog node, the TEE verifies the signature of the received packet, loads the monitor and Target_TA, allocates their own memory space for them, and provides the use of the TEE's internal APIs to access key material and sensor data. In the real world, a CA is deployed in parallel, operating in a separate security domain with minimally required privileges – access only to the necessary sensors, network interfaces, and the TEE client API. This is achieved by using mandatory access control mechanisms that use the "default denied" model, such as SELinux [33]. This allows setting a policy for the CA that blocks direct access to secrets, TA file stores, and critical system resources. As a result, even if the CA is compromised, the attacker is limited to the scope of permitted operations and cannot directly affect the logic or confidentiality of processing inside the TEE.

The transfer of sensitive data and keys in the framework is also separated. A prerequisite for key transfer is the mandatory attestation of the TA [23]. The sequence of steps for local evidence collection, attestation of the target Trusted Application after its launch, and confidential distribution of secrets is shown in Fig. 7. First, the normal world receives platform evidence (TPM Quote), and the TEE generates a TA_Report with a hash of the code, version, and a one-time public key; these two types of evidence are combined and verified by the mobile verifier against reference RIMs and a known "good" TA binary image. Only after a successful double-check does the mobile verifier encrypt the secret material (keys for encrypting medical data, keys for state, etc.) with a one-time public key embedded in the Target_TA report and send the encrypted container to the fog node. Decryption occurs inside the TEE using the corresponding private key, after which the secrets are stored in a secure per-TA storage and never appear in the regular environment. The CA itself works with the data as encrypted payloads and sees only the minimal metadata needed for routing and orchestrating tasks.

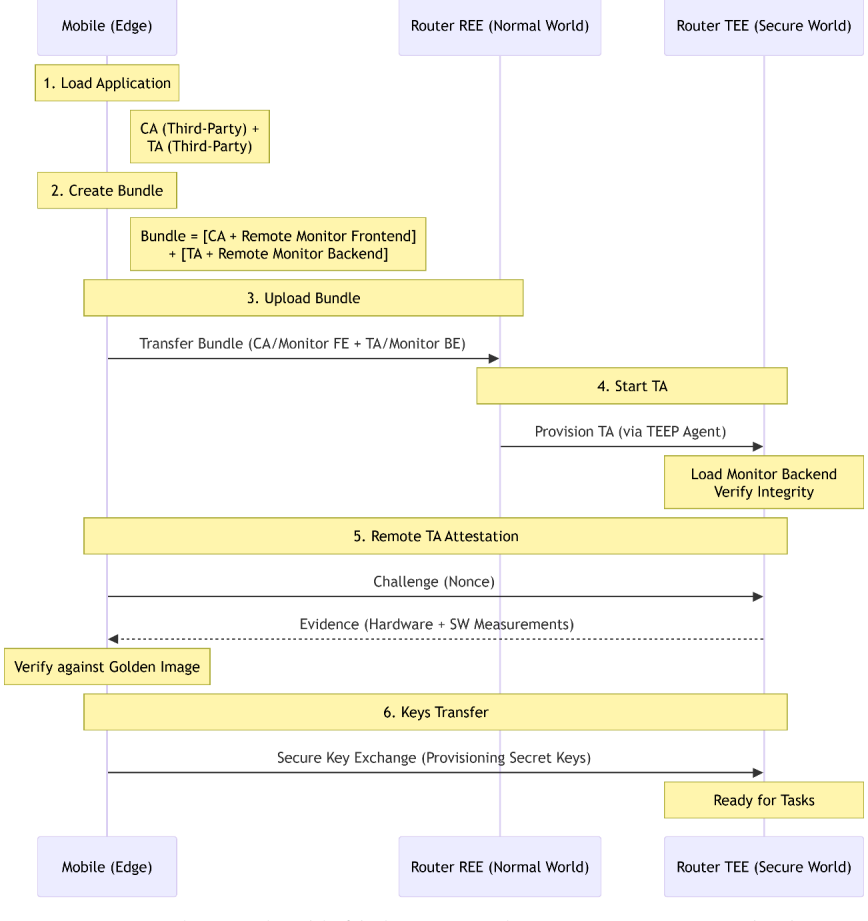

Fig. 4. Logical-structural model of deploying user applications on a remote untrusted node

ISSN-L 2664-9969; E-ISSN 2706-5448

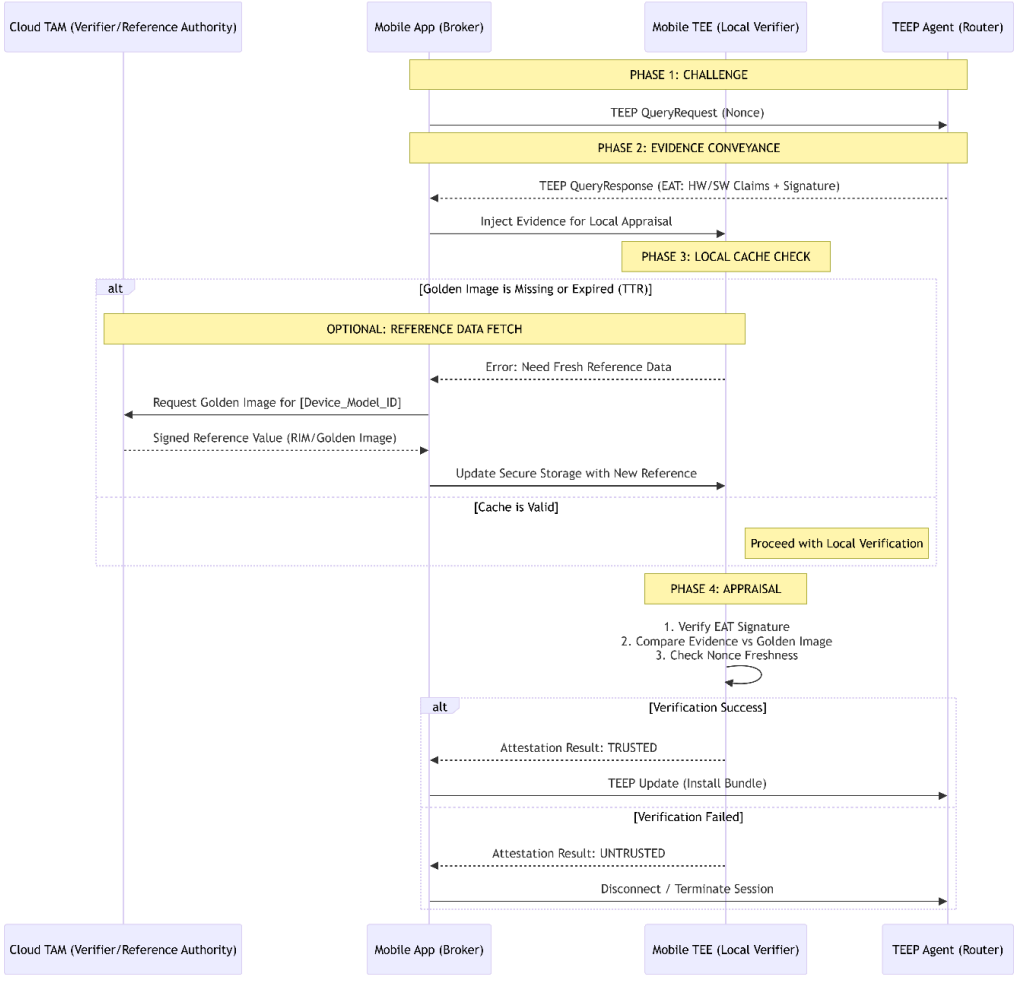

Fig. 5. The process of attesting an untrusted remote device

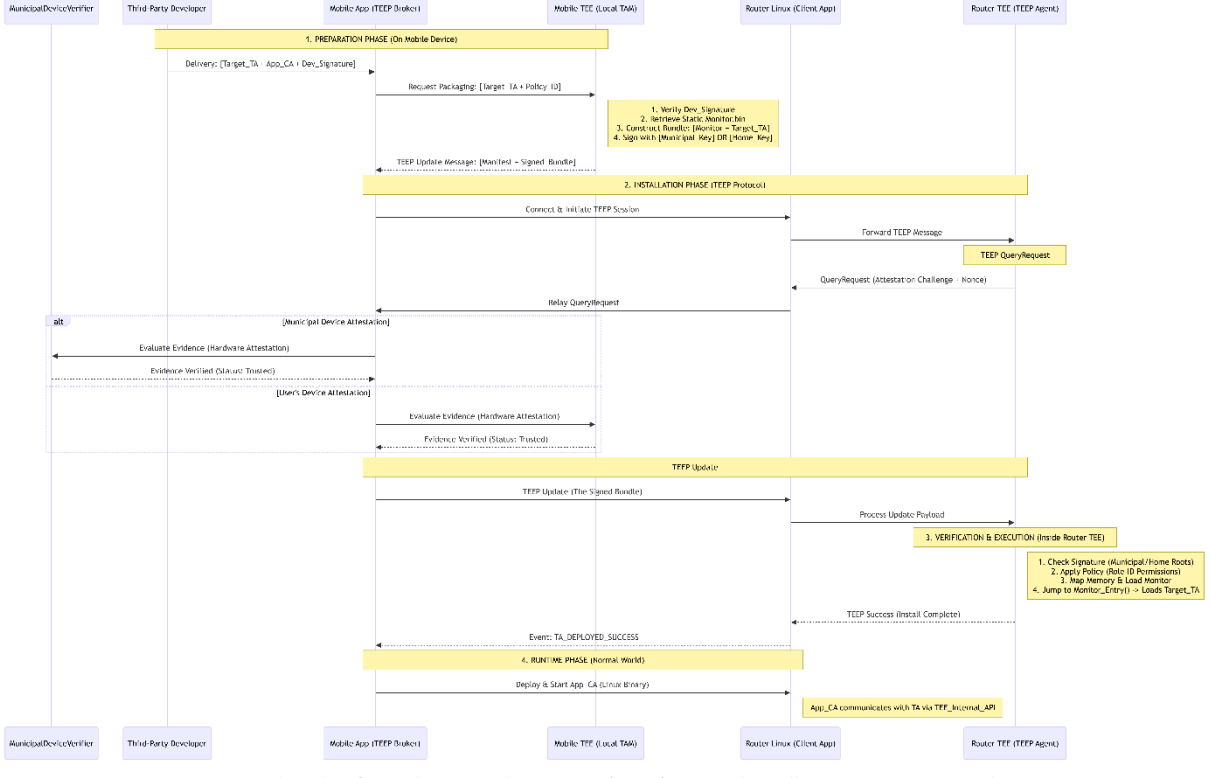

Fig. 6. Algorithm for implementing the process of transferring and installing TA on a remote node

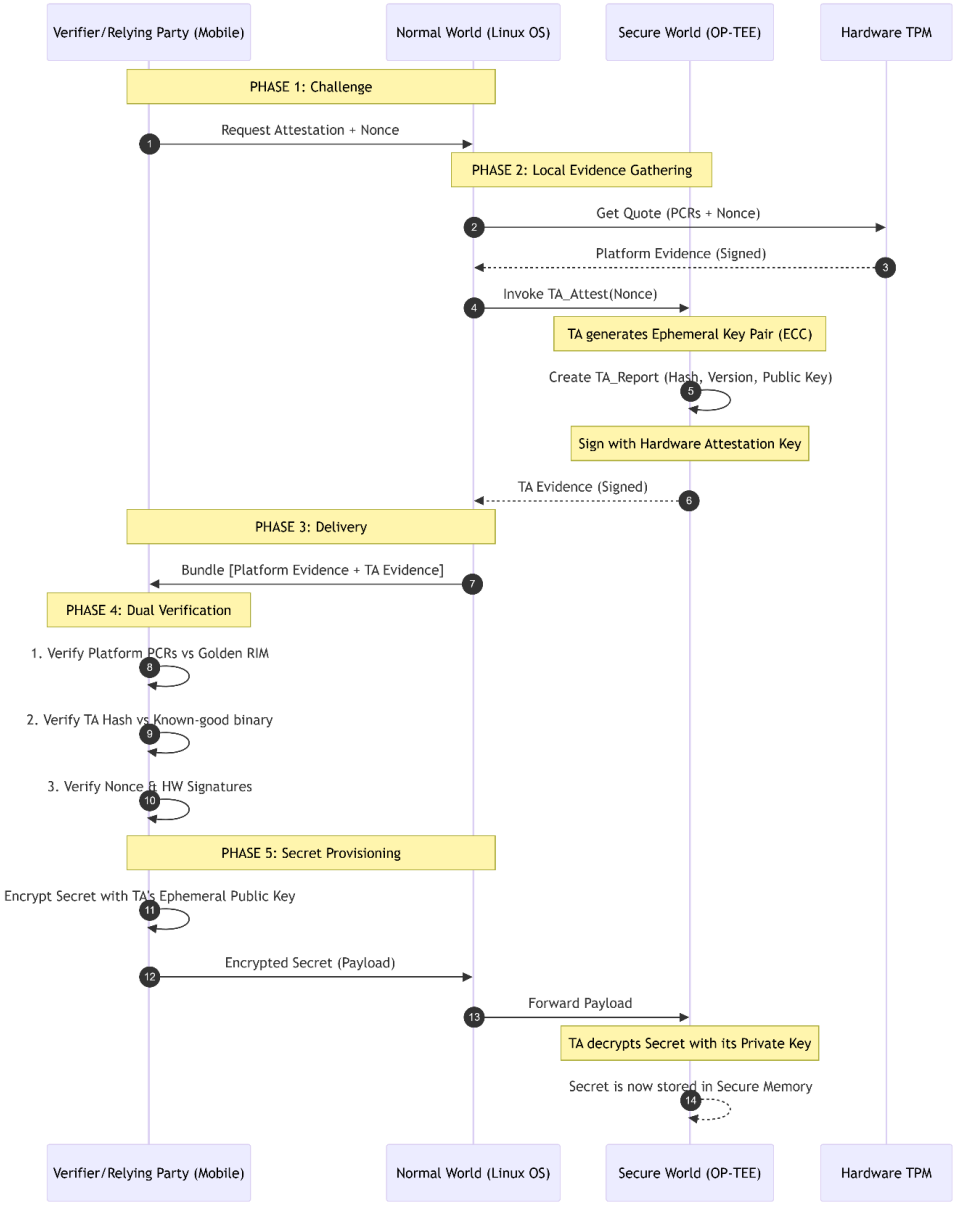

Fig. 7. Certification of user TA for cryptographic key transfer

The proposed confidential execution framework is integrated with the system task placement model as an additional layer of assumptions about trust in fog nodes. In the subsequent performance analysis (latency, energy consumption), it is assumed that the nodes selected to perform RPM tasks satisfy the certification requirements and perform data processing inside the TEE with minimal privileges in the surrounding system. A detailed quantitative assessment of the overhead of certification procedures, TEE initialization and SELinux security policies is beyond the scope of this work and is considered as a direction for further research.

3.3. Simulation setup, scenarios, and validation

3.3.1. Simulation environment in Yet Another Fog Simulator (YAFS)

To quantitatively evaluate the proposed model, the discrete-event simulator YAFS [34] was used, which supports the modeling of multi-level fog architectures, network topologies, and task distribution between nodes. The fog node parameters of the PCEngines ALIX 3D2 node, the Samsung Galaxy S4 smartphone, and the LG Urbane smart watch were used as the hardware basis in the simulation. An extended model of energy consumption and the Wi-Fi channel was integrated into the basic functionality of YAFS: the dependence of the transmission energy on the data volume and RSSI, the possibility of retransmissions when the channel quality deteriorates, and the drop in the effective transmission speed over long distances were taken into account. For BLE connections, the simulation assumed a stable channel (short range of wearable devices), so their contribution to energy consumption is modeled as a constant component.

The topology consists of one mobile user (edge node, smartphone), one fog node (home/city gateway), and a cloud node. Sensors (ECG device and smart watch (also acts as a temperature and humidity sensor) are connected to the edge or fog layer, depending on the scenario. To assess the impact of radio channel conditions, several *RSSI* values for Wi-Fi connection are considered: –40 dBm, –70 dBm, –85 dBm, corresponding to good, average, and poor signal quality. To model the impact of wireless channel quality, empirical profiles were used, built on the basis of the results of work [29]. The dependence of Wi-Fi throughput on *RSSI* is given by a piecewise linear approximation (Fig. 8): at *RSSI* from –50 to –70 dBm, the effective throughput remains at about 54 Mbps (802.11ag), while further reduction of the signal to –75 dBm, –85 dBm and –90 dBm leads to a drop in speed to 11, 11 and 1 Mbps, respectively, and at –95 dBm channel is practically unavailable.

ISSN-L 2664-9969; E-ISSN 2706-5448

For TCP traffic of tasks *T1, T3, T4, T5*, the dependence of the rate of retransmissions on *RSSI* is introduced (Fig. 9): at –60 dBm there are no retransmissions, at –70 dBm the rate reaches about 20%, at –80 dBm – 30%, and in the range from –90 to –95 dBm it exceeds 90%. This directly affects both the delay and the power consumption of the wireless interface.

The power consumption of the Wi-Fi interface of a smartphone (Samsung Galaxy S4) [35] is modeled as a function of RSSI with separate profiles for transmission and reception (Fig. 10). At –50 dBm, the transmit/receive power is approximately 654/451 mW, increasing to 1113/633 mW at –80 dBm, after which it decreases to 671/395 mW at –90 dBm due to a sharp drop in throughput.

Similar profiles are built for the Wi-Fi interface of the smart watch (Fig. 11): the transmit power increases from ≈669 mW at –42 dBm to ≈841 mW at –65...–70 dBm, while the receive power decreases from ≈379 to ≈252 mW.

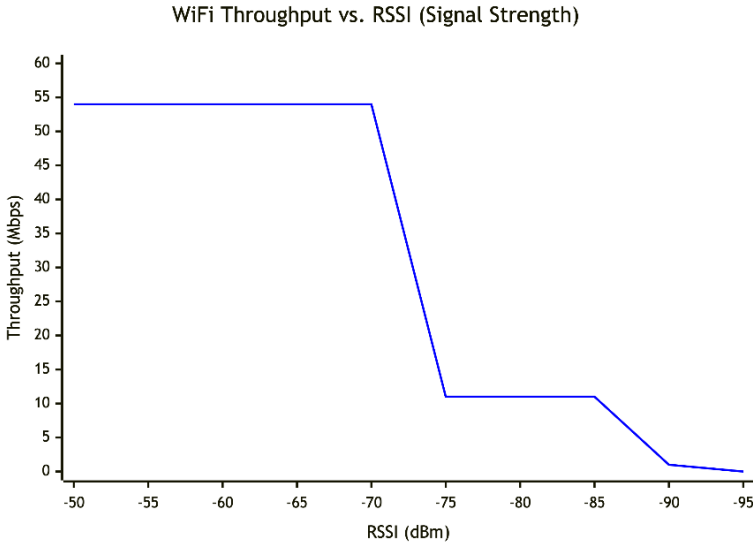

Fig. 8. Dependence of Wi-Fi channel bandwidth on RSSI signal strength level indicator

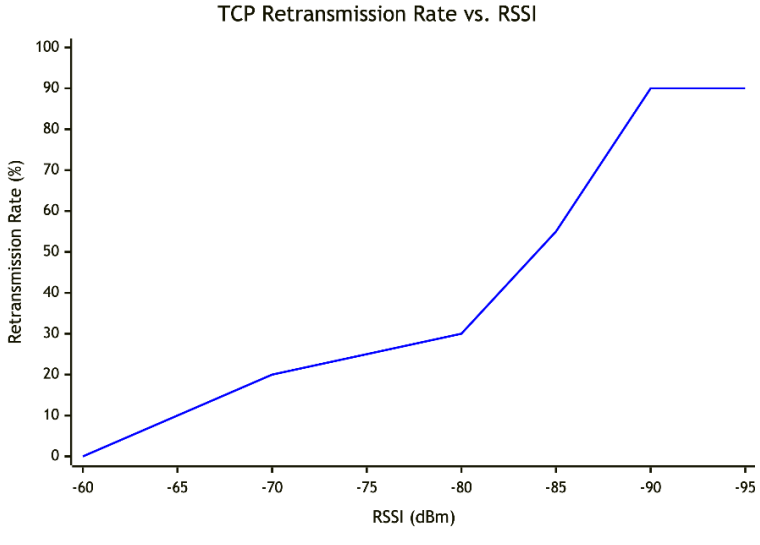

Fig. 9. Dependence of the percentage of TCP packet retransmission over the Wi-Fi channel on RSSI signal strength level indicator

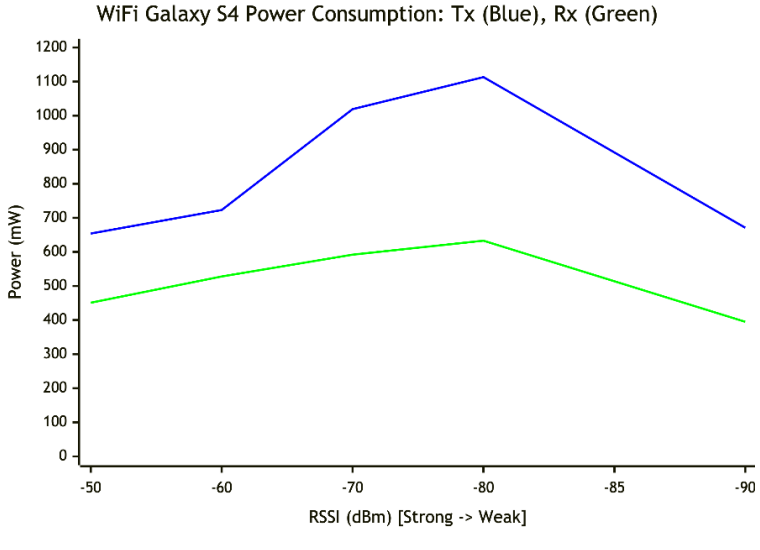

Fig. 10. Dependence of the transmit and receive power of the Wi-Fi interface of the Samsung Galaxy S4 phone on the RSSI signal strength level indicator

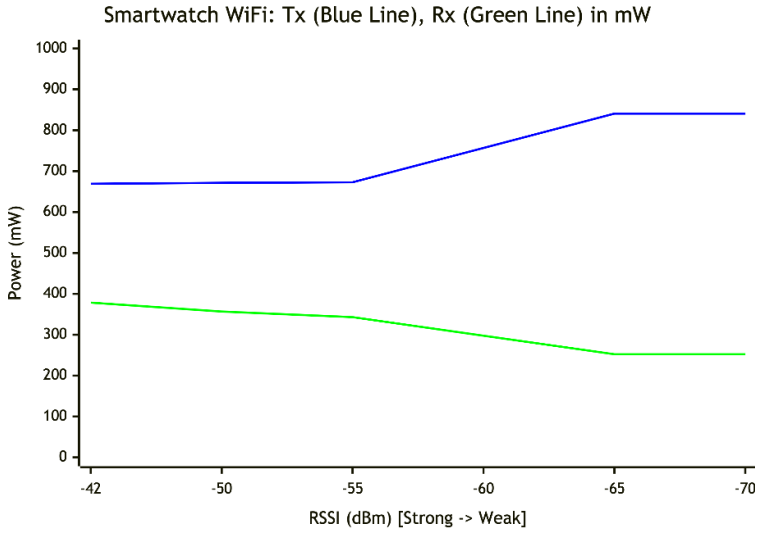

Fig. 11. Dependence of the transmission and reception power of the Wi-Fi interface of the LG Urbane Watch smart watch on the RSSI signal strength level indicator

These dependencies are used by the simulator to calculate the instantaneous power during frame transmission and reception, depending on the current RSSI. They also explain the observed sensitivity of the mobile scenario to RSSI degradation compared to the fog scenario, in which the transmission path for most tasks is shorter.

A practical feature of smart watches was separately taken into account: according to the results of [36], when the Wi-Fi signal level dropped below ≈–70 dBm, regular problems with maintaining the connection and frequent session breaks were observed for a number of models. In view of this, simulation experiments for the hybrid and fog scenarios were performed only in the RSSI range up to –70 dBm; for worse signal values, these configurations are considered unsuitable for reliable operation and are not analyzed in this work.

3.3.2. Remote patient monitoring (RPM) task set

To validate the proposed model, a representative set of Remote Patient Monitoring (RPM) tasks was selected, reflecting typical computational workloads in modern e-health scenarios [37]. The characteristics of these tasks are given in Table 1:

– task ID (unique computational task identifier, *T1*–*T5*);
– load type/scenario (functional purpose of the task, e. g., ECG classification, voice audio signal analysis);
– period, *P* (time interval between generation of new data packets by sensors, i. e., the computational task execution period);
– computational complexity, *C* (the amount of processor resources required for processing, expressed in millions of instructions, Million Instructions, MI);
– input data volume, D_{in} (size of the raw data packet coming from the sensor layer);
– time limit/deadline (maximum allowable task execution time, including transmission and processing);
– output volume, D_{out} (size of the result after task execution).

In Fog and Hybrid scenarios, output data D_{out} also acts as "control points" (state snapshots), which allows to resume calculations or to transfer the final result to the cloud in case of loss of connection.

The determination of output data volumes (D_{out}) depends on the compression and feature extraction algorithms. For task *T1* (ECG classification), the output data volume is approximately 2% of the input data volume due to the use of the Dynamic Time Warping (DTW) algorithm [38]. For audio analysis (*T2*), it is possible to assume that the output vector includes only key acoustic descriptors (perceptual loudness, zero-crossing rate, spectral centroid, and instantaneous energy), which makes its size no more than 4 kB [39]. The environmental monitoring tasks (*T3*, *T4*) demonstrate the highest compression ratio (up to 99.93%) due to the low entropy of temperature and humidity indicators [40] over time. This allows removing the constant component from the data and transmitting only the deltas of the indicator changes. For the ECG compression task (*T5*), the data reduction ratio is 93%, according to the performance indicators [41].

Each task is modeled as a deterministic periodic flow of requests. The trajectory of the request is determined by the selected placement strategy (Edge-centric, Fog-centric or Hybrid). After the computation phase is completed on the corresponding node, the fixed-size results are transmitted to the cloud storage for long-term analytics and to the user's mobile device for real-time visualization of the current health status.

The simulation uses real-world device parameters taken from published power consumption measurements:

– *Fog node* is modeled on a PCEngines ALIX 3D2 platform (500 MHz x86 [42], 256 MB RAM) with 500 MIPS performance, 0.9 W CPU power consumption, and 4.9 W (transmit)/3.7 W (receive) radio power [43];
– *Mobile device* matches the specifications of a Galaxy S4 smartphone (1.9 GHz Quad-Core Krait 300, 2 GB RAM): 1900 MIPS, 1.3 W CPU power consumption at maximum frequency for one core, Wi-Fi transmit/receive power of 654/451 mW at –50 dBm and 1113/633 mW at –80 dBm, and BLE 4.0 with 174 mW receive power. The duration of the final (tail) part of the Wi-Fi session is 210 ms, with a fixed power of 289 mW. The duration of the tail part of the BLE transmission is taken from the characteristics of the smart watch, due to the similarity of the BLE standards – 4.77 s, 34.1 mW;
– *Sensor watch* is modeled according to the parameters of the LG Urbane (768 MHz Quad-Core Cortex-A7, 512 MB RAM): 768 MIPS, CPU power consumption 361 mW, Wi-Fi power 739.9/400.1 mW (transmit/receive), and BLE 4.1 with power 180.7/174.9 mW. The duration of the tail part of the BLE is 4.77 s, 34.1 mW [37]. The ECG tracker in the simulation is considered identical to the watch in terms of power characteristics.

The sources for these parameters are publications on the power consumption of smartphones and wearable devices, and power measurements of Wi-Fi/BLE interfaces. For BLE 4.0/4.1, the model uses a realistic maximum throughput of about 0.305 Mbps, which is significantly less than the nominal value of 1 Mbps and is consistent with experimental data [44].

ISSN-L 2664-9969; E-ISSN 2706-5448

3.3.3. Scenarios description: mobile, hybrid, and fog

Scenario 1 (Mobile scenario): All tasks *T1–T5* are performed at the edge level, i. e. on the user's smartphone. Wearable sensors (ECG, smart watch) are connected to the smartphone via BLE, and the processing results are transmitted to the fog node via Wi-Fi, and then to the cloud. Thus, the mobile device carries the main computational load, and the Wi-Fi channel is used mainly for transmitting the results.

Scenario 2 (Hybrid scenario): The smart watch sends data to the smartphone via BLE, but then the "raw" telemetry without pre-processing is transmitted to the fog node via Wi-Fi, where all tasks *T1–T5* are performed. The ECG sensor in this scenario is connected directly to the fog node via Wi-Fi. The processing results are returned to the smartphone (for storing the latest state and local visualization) and are sent to the cloud in parallel. The computational load is shifted from the edge to the fog, but the mobile device is still actively transmitting significant amounts of data over Wi-Fi.

Scenario 3 (Fog scenario): All sensors (ECG, smartwatch, environmental sensors) are connected directly to the fog node, where all tasks *T1–T5* are performed. The smartphone receives only the processing results necessary for displaying the state, while the full set of results is transmitted to the cloud for long-term storage and analytics. In this scenario, the mobile device performs almost no calculations and uses the Wi-Fi channel minimally, which reduces power consumption and network latency on its side.

For each combination of scenario and RSSI, the following metrics are measured in the simulation:

– mean latency – the time from the moment the task is generated to the result on the processing node;
– mean service time – the net calculation time on the selected node without taking into account transmission;
– mobile device power consumption, divided into the link and local computing (service) components;
– wearable sensor power consumption for communication;
– total network load (volume of transmitted data).

These metrics directly correspond to the time $T(X)$ and energy $E(X)$ model components. They allow to check whether the system behavior in static scenarios is consistent with intuitive expectations. In particular, whether the fog scenario really reduces the latency and power consumption of the mobile device compared to the mobile scenario at different RSSI values.

3.3.4. Performance analysis and assessment of the impact of fog computing on latency and energy consumption RPM

The total simulation time for each scenario was 1 hour. In the mobile scenario, the average latency for all tasks is approximately 0.57 s and changes slightly when the signal deteriorates from –40 to –85 dBm (0.56866 s, 0.568822 s, 0.574825 s, respectively), since the main contribution is made by local computing on the smartphone. In the hybrid scenario, the average latency decreases to 0.384–0.387 s due to the transfer of processing to the fog node, although part of the traffic still passes through the smartphone. According to empirical data [37], at RSSI <–70 dBm, the Wi-Fi of a smart watch begins to communicate poorly, which is why the corresponding power measurements were not made, so there is no data for the simulation of the hybrid and fog scenarios. Table 2 presents the summarized average results of the scenario simulation at different RSSI levels.

The fog scenario demonstrates the lowest latency: the average latency for all tasks is 0.0267 s at RSSI –40 dBm and increases to 0.0296 s at –70 dBm, i. e., it remains almost an order of magnitude lower than in the mobile configuration. At the same time, the average task service time (CPU part) in the hybrid and fog scenarios is the same (0.6924 s), which reflects the lower computing power of the fog node compared to the Galaxy S4 smartphone. The fog scenario's advantage is due to the reduction of the network component of the path – the sensors are directly connected to the fog. The total execution time in the fog and hybrid configurations remains comparable and can be further reduced when using more powerful fog devices.

The energy consumption of a mobile phone differs significantly between the scenarios. In the mobile scenario, the smartphone simultaneously performs all tasks and acts as the main network node, so the total energy consumption (communication + computing) reaches approximately 222–225 mWh per simulation interval (104.9–107.6 mWh for transmission/reception + 117 mWh for local service). In the hybrid scenario, where the calculations are transferred to the fog layer, the service energy on the smartphone becomes zero, and the energy for communication decreases to 74.6–77.5 mWh; thus, the savings relative to the mobile scenario are about 2.5–3 times.

Table 2

RPM task performance in three architectural scenarios (mobile, hybrid, fog) at different RSSI values

Scenario and characteristics	*Value RSSI* = –40 dBm	*Value RSSI* = –70 dBm	*Value RSSI* = –85 dBm
Mobile scenairo			
Network load	288.46 MiB	290.26 MiB	296.18 MiB
Mobile Energy consumption (Link + Service)	104.9 + 117 mWh	105.1 + 117 mWh	107.6 + 117 mWh
Sensor Energy consumption (Link)	83 mWh	83 mWh	83 mWh
Mean latency (all tasks)	0.56866 s	0.568822 s	0.574825 s
Mean time service (all tasks)	0.182211 s	0.182211 s	0.182211 s
Hybrid scenario			
Network load	464.90 MiB	489.93 MiB	–*
Mobile Energy consumption (Link + Service)	74.6 + 0 mWh	77.5 + 0 mWh	–
Sensor Energy consumption (Link)	47.4 mWh	49 mWh	–
Mean latency (all tasks)	0.384403 s	0.387024 s	–
Mean time service (all tasks)	0.6924 s	0.6924 s	–
Fog scenario			
Network load	288.46 MiB	316.47 MiB	–
Mobile Energy consumption (Link + Service)	20.3 + 0 mWh	20.4 + 0 mWh	–
Sensor Energy consumption (Link)	15.13 mWh	18.1 mWh	–
Mean latency (all tasks)	0.026717 s	0.029593 s	–
Mean time service (all tasks)	0.6924 s	0.6924 s	–

The greatest gain is achieved in the fog scenario: the smartphone transmits only the results from the fog node and does not perform calculations, so the energy for communication decreases to ≈20.3–20.4 mWh, and the energy for service remains zero. Compared to the mobile configuration, this reduces the smartphone's power consumption by more than ten times. This behavior is consistent with the energy profiles of the Galaxy S4 Wi-Fi interface: when RSSI deteriorates, the transmit and receive power increase to more than 1 W, and the throughput decreases, which increases the radio module's activity time and the total energy consumed.

For wearable sensors (smartwatch and ECG tracker), the power consumption for communication also depends on the scenario and RSSI, but the changes are of a smaller order of magnitude. In the mobile scenario, the sensors transmit data to the smartphone via BLE, where the power consumption is relatively stable; in the hybrid and fog scenarios, a Wi-Fi channel appears to the fog node, for which separate power profiles are built. In the simulation results, the total energy of the sensors per connection changes from ≈83 mWh in the mobile scenario to 47–49 mWh in the hybrid scenario and 15–18 mWh in the fog scenario. This shows additional savings when reducing the length of the wireless path and the number of retransmissions.

The network load shows the expected differences between the scenarios. In the mobile and fog scenarios, where the bypass of raw data through the smartphone is minimized, the total amount of transmitted data is about 288–316 MiB (depending on RSSI). In contrast, in the hybrid scenario, where the smartphone sends the full telemetry streams to the fog layer, the network load increases to 465–490 MiB. This confirms that the fog scenario combines the advantages of low latency and low power consumption with moderate network load, while the hybrid is an intermediate compromise between the mobile and fog configurations.

The results obtained demonstrate that the proposed task placement model and the modernized simulator correctly reflect the expected dependence between architectural scenarios, wireless channel quality, and energy and time characteristics of mobile devices. The Fog scenario provides the lowest network latency by executing tasks directly on the fog node connected to the sensors, and at the same time minimizes the smartphone's power consumption, since it does not perform calculations and only transmits results. The Mobile scenario is the worst in both indicators. Even with a good RSSI level, the phone is forced to spend significant energy on Wi-Fi communication and processing all tasks locally. When the signal deteriorates and the bandwidth drops, additional retransmissions increase costs even more.

The hybrid configuration is an intermediate option: it reduces the smartphone's power consumption by offloading computations to the fog layer, but creates additional network load by sending raw sensor data through the smartphone to the fog node. From the perspective of the $\alpha E(X) + \beta T(X)$ model, the hybrid scenario corresponds to a compromise setup, while the fog scenario implements a mode with high weight on latency and energy of mobile devices. This is consistent with the trends of modern offloading work, where the fog layer is considered a key component for ensuring QoS in e-health systems. It is important that the estimates obtained for the fog scenario are conservative: the simulation used a fog node with less computing power than the smartphone, so the task service time turned out to be longer than in the mobile scenario. In real-world deployments, where the fog infrastructure may include modern x86/ARM microservers, it is possible to expect further reductions in overall latency without changing the task placement principles. In addition, the proposed framework for confidential execution on fog nodes allows to combine these latency and energy gains with guarantees of confidentiality and integrity of processing on partially untrusted nodes, which is critical for implementing RPM in a smart city without using separate secure nodes.

The results obtained are consistent with the general conclusions of previous works on the advantages of edge/fog architectures for RPM, but extend them by using a more detailed energy model and taking into account the quality of Wi-Fi and BLE channels. In contrast to the approaches [13–18], where offloading is analyzed mostly on abstract scenarios and without modeling the energy consumption of mobile devices, this work considers a detailed model of the energy consumption of a smartphone. It is shown that the transition from mobile to fog scenario allows to significantly reduce the average and reduce the power consumption of the smartphone, which quantitatively confirms the feasibility of fog-centric solutions for streaming medical tasks. Compared to the works [19–21], which mainly focus on the architectural aspects of RPM and qualitative comparison of edge/fog models, the proposed model also integrates confidentiality requirements. They are implemented through TEE and remote attestation, which expands the scope of practical application of fog infrastructure in a smart city.

3.4. Research limitations and directions of its development

The limitations of this research are the use of a simplified simulation topology with one user and one fog node, fixed static task placement policies and approximate energy consumption models of BLE interfaces. As well as the absence in the simulation of overhead associated with TEE initialization, remote attestation procedures, and implementation of security policies at the fog level, as well as offloading of the computational task.

Further research directions are the development and analysis of adaptive offloading algorithms to minimize the weighted trade-off criterion between mobile device energy consumption and time delay $J(X)$ under variable network conditions and load. The second direction is to scale the model to multi-user scenarios with multiple fog nodes. The third direction is the quantitative assessment of the impact of TEE/RA overhead and forced access control mechanisms on latency and energy consumption in real prototypes of RPM systems.

4. Conclusions

1. A system model for dynamic distribution of remote health monitoring (RPM) tasks in a multi-level Cloud-Fog-Edge infrastructure is formulated. The model is based on an integrated energy-time criterion (Weighted Energy-Delay Product), which allowed formalizing the offloading process as a multi-criteria optimization problem. The use of the model ensures that delays and strict deadlines of medical services are taken into account, which allows adapting the resource allocation strategy to the critical state of the patient in real time.

2. A conceptual framework for secure execution of tasks on peripheral nodes is proposed, which, for the first time, integrates Trusted Execution Environment (TEE) technologies and remote attestation mechanisms (RATS) into medical offloading processes. A qualitative indicator of the result is the implementation of the principle of least privilege (PoLP) at the Fog node level, which guarantees the integrity and confidentiality of biometric data even in a public environment uncontrolled by the user. This eliminates the risks of data compromise at the node operating system level and ensures the system's compliance with regulatory requirements for the protection of medical information.

3. An experimental verification of the developed model was carried out in the YAFS simulation environment using refined energy consumption profiles and wireless channel characteristics. Based on the testing of three scenarios (Mobile, Hybrid, Fog), the effectiveness of the proposed approach was quantitatively confirmed. It was found that the transition from Mobile- to Fog-oriented strategy provided a decrease in the average latency of the system from 0.57 s to 0.027–0.030 s. It was proven that the power consumption of the mobile terminal decreased from 222–225 mWh to 20.3–20.4 mWh (more than 10 times), and the load on the battery of wearable sensors decreased from 83 mWh to 15–18 mWh. (almost 5 times). It was found that even when using Fog nodes with lower computing power (500 MIPS versus 1900 MIPS for

ISSN-L 2664-9969; E-ISSN 2706-5448

a smartphone), the architectural advantages of fog computing provide a significant improvement in QoS by minimizing the network component. The results obtained confirmed the high stability of the model to equipment heterogeneity and proved the feasibility of transferring computing to the Fog level for critical medical services within a smart city.

Conflict of interest

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in this paper.

Financing

The research was performed without financial support.

Data availability

Manuscript has no associated data.

Use of artificial intelligence

During the preparation of this work, the authors used Grammarly to Perform Grammar and spelling checks. After using this service, the authors reviewed and edited the content as needed and took full responsibility for the publication's content.

Authors' contributions

Ivan Shevtsov: Conceptualization, Methodology, Validation, Investigation, Data curation, Writing – original draft, Writing – review and editing; ***Tetiana Fesenko***: Conceptualization, Methodology, Formal analysis, Resources, Writing – original draft, Writing – review and editing, Supervision.

References

1. Farias, F. A. C., Dagostini, C. M., Bicca, Y. A., Falavigna, V. F., Falavigna, A. (2020). Remote Patient Monitoring: A Systematic Review. *Telemedicine and E-Health, 26 (5),* 576–583. https://doi.org/10.1089/tmj.2019.0066
2. Malasinghe, L. P., Ramzan, N., Dahal, K. (2017). Remote patient monitoring: a comprehensive study. *Journal of Ambient Intelligence and Humanized Computing, 10 (1),* 57–76. https://doi.org/10.1007/s12652-017-0598-x
3. Dadkhah, M., Mehraeen, M., Rahimnia, F., Kimiafar, K. (2021). Use of Internet of Things for Chronic Disease Management. *Journal of Medical Signals & Sensors, 11 (2),* 138–157. https://doi.org/10.4103/jmss.jmss_13_20
4. HealthTrack SG (2026). *Health Promotion Board.* Available at: https://www.hpb.gov.sg/healthy-living/healthtracksg
5. Nanehkaran, Y. A., Licai, Z., Chen, J., Zhongpan, Q., Xiaofeng, Y., Navaei, Y. D. et al. (2022). Diagnosis of Chronic Diseases Based on Patients' Health Records in IoT Healthcare Using the Recommender System. *Wireless Communications and Mobile Computing, 2022 (1).* https://doi.org/10.1155/2022/5663001
6. Rodrigues, V. F., da Rosa Righi, R., da Costa, C. A., Zeiser, F. A., Eskofier, B., Maier, A. et al. (2023). Digital health in smart cities: Rethinking the remote health monitoring architecture on combining edge, fog, and cloud. *Health and Technology, 13 (3),* 449–472. https://doi.org/10.1007/s12553-023-00753-3
7. *Pro zatverdzhennia normatyvnykh dokumentiv shchodo zastosuvannia telemedytsyny u sferi okhorony zdorovia* (2015). Nakaz MOZ Ukrainy No. 681. 19.10.2015. Available at: https://zakon.rada.gov.ua/go/z1400-15 Last accessed: 03.02.2026
8. Byshenko, H., Avtomieienko, Y. (2024). Analysis of the government policy of the reform of electronic health care and medicine of Ukraine. *State Formation, 1 (35),* 290–304. https://doi.org/10.26565/1992-2337-2024-1-22
9. *OpenFog Reference Architecture for Fog Computing* (2017). OpenFog Consortium, 162. Available at: https://www.iiconsortium.org/pdf/OpenFog_Reference_Architecture_2_09_17.pdf
10. Muneeb, M., Ko, K.-M., Park, Y.-H. (2021). A Fog Computing Architecture with Multi-Layer for Computing-Intensive IoT Applications. *Applied Sciences, 11 (24),* 11585. https://doi.org/10.3390/app112411585
11. Hossam, H. S., Abdel-Galil, H., Belal, M. (2024). An energy-aware module placement strategy in fog-based healthcare monitoring systems. *Cluster Computing, 27 (6),* 7351–7372. https://doi.org/10.1007/s10586-024-04308-7
12. Mahmoud, M. M. E., Rodrigues, J. J. P. C., Saleem, K. (2019). Cloud of Things for Healthcare: A Survey from Energy Efficiency Perspective. *2019 International Conference on Computer and Information Sciences (ICCIS).* Sakaka: IEEE, 1–7. https://doi.org/10.1109/iccisci.2019.8716388
13. Dong, S., Tang, J., Abbas, K., Hou, R., Kamruzzaman, J., Rutkowski, L. et al. (2024). Task offloading strategies for mobile edge computing: A survey. *Computer Networks, 254,* 110791. https://doi.org/10.1016/j.comnet.2024.110791
14. Matrouk, K., Alatoun, K. (2021). Scheduling Algorithms in Fog Computing: A Survey. *International Journal of Networked and Distributed Computing, 9 (1),* 59. https://doi.org/10.2991/ijndc.k.210111.001
15. Adhikari, M., Gianey, H. (2019). Energy efficient offloading strategy in fog-cloud environment for IoT applications. *Internet of Things, 6,* 100053. https://doi.org/10.1016/j.iot.2019.100053
16. Fan, J., Liu, J., Chen, J., Yang, J. (2018). LPDC: Mobility-and Deadline-Aware Task Scheduling in Tiered IoT. *2018 IEEE 4th International Conference on Computer and Communications (ICCC).* Chengdu: IEEE, 857–863. https://doi.org/10.1109/compcomm.2018.8780904
17. Gao, X., Huang, X., Bian, S., Shao, Z., Yang, Y. (2020). PORA: Predictive Offloading and Resource Allocation in Dynamic Fog Computing Systems. *IEEE Internet of Things Journal, 7 (1),* 72–87. https://doi.org/10.1109/jiot.2019.2945066
18. Li, C., Tang, J., Zhang, Y., Yan, X., Luo, Y. (2019). Energy efficient computation offloading for nonorthogonal multiple access assisted mobile edge computing with energy harvesting devices. *Computer Networks, 164,* 106890. https://doi.org/10.1016/j.comnet.2019.106890
19. Shevtsov, I. (2024). Actual problems of remote patient monitoring. *Computer-Integrated Technologies: Education, Science, Production, (56),* 5–11. https://doi.org/10.36910/6775-2524-0560-2024-56-01
20. Shevtsov, I. (2024). The comparative analysis of the effectiveness of edge computing and fog computing in medical monitoring systems. *Information Technology and Society, 3 (14),* 44–53. https://doi.org/10.32689/maup.it.2024.3.6
21. Shevtsov, I. O. (2024). Hybrid computing models (fog and edge) for optimizing remote monitoring of chronic diseases. *Scientific Notes of Taurida National V. I. Vernadsky University. Series: Technical Sciences, 1 (5),* 343–353. https://doi.org/10.32782/2663-5941/2024.5.1/48
22. *TEE Internal Core API Specification.* GlobalPlatform. Available at: https://globalplatform.org/wp-content/uploads/2021/03/GPD_TEE_Internal_Core_API_Specification_v1.3.1_PublicRelease_CC.pdf
23. *RFC 9334: Remote ATtestation procedureS (RATS) Architecture.* IETF Datatracker. Available at: https://datatracker.ietf.org/doc/rfc9334/
24. Ménétrey, J., Göttel, C., Khurshid, A., Pasin, M., Felber, P., Schiavoni, V. et al. (2022). Attestation Mechanisms for Trusted Execution Environments Demystified. *Distributed Applications and Interoperable Systems.* Cham: Springer, 95–113. https://doi.org/10.1007/978-3-031-16092-9_7
25. Albahri, O. S., Albahri, A. S., Mohammed, K. I., Zaidan, A. A., Zaidan, B. B., Hashim, M. et al. (2018). Systematic Review of Real-time Remote Health Monitoring System in Triage and Priority-Based Sensor Technology: Taxonomy, Open Challenges, Motivation and Recommendations. *Journal of Medical Systems, 42 (5).* https://doi.org/10.1007/s10916-018-0943-4
26. Kraemer, F. A., Braten, A. E., Tamkittikhun, N., Palma, D. (2017). Fog Computing in Healthcare – A Review and Discussion. *IEEE Access, 5,* 9206–9222. https://doi.org/10.1109/access.2017.2704100
27. Skorin-Kapov, L., Matijasevic, M. (2010). Analysis of QoS Requirements for e-Health Services and Mapping to Evolved Packet System QoS Classes. *International Journal of Telemedicine and Applications, 2010,* 1–18. https://doi.org/10.1155/2010/628086
28. Gallego, J. R., Hernandez-Solana, A., Canales, M., Lafuente, J., Valdovinos, A., Fernandez-Navajas, J. (2005). Performance analysis of multiplexed medical data transmission for mobile emergency care over the UMTS channel. *IEEE Transactions on Information Technology in Biomedicine, 9 (1),* 13–22. https://doi.org/10.1109/titb.2004.838362
29. Ding, N., Wagner, D., Chen, X., Pathak, A., Hu, Y. C., Rice, A. (2013). Characterizing and modeling the impact of wireless signal strength on smartphone battery drain. *Proceedings of the ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems,* 29–40. https://doi.org/10.1145/2465529.2466586
30. Arm Firmware Framework for Arm A-profile (2025). *Arm Ltd.* Available at: https://developer.arm.com/-/cdn-downloads/permalink/Architectures/Armv9/DEN0077A_Firmware_Framework_Arm_A-profile_1.3_ALP1_ALP2_Diff.pdf
31. Pei, M., Tschofenig, H., Thaler, D. (2023). Trusted Execution Environment Provisioning (TEEP) Architecture. *Internet Engineering Task Force.* Available at: https://datatracker.ietf.org/doc/html/rfc9397

32. Remote Integrity Verification of Network Devices Containing Trusted Platform Modules (2024). *IETF.* Available at: https://datatracker.ietf.org/doc/rfc9683/ Last accessed: 27.03.2026
33. *Security-Enhanced Linux in Android.* Available at: https://source.android.com/docs/security/features/selinux/ Last accessed: 27.03.2026
34. Lera, I., Guerrero, C., Juiz, C. (2019). YAFS: A Simulator for IoT Scenarios in Fog Computing. *IEEE Access, 7,* 91745–91758. https://doi.org/10.1109/access.2019.2927895
35. Chen, X., Ding, N., Jindal, A., Hu, Y. C., Gupta, M., Vannithamby, R. (2015). Smartphone Energy Drain in the Wild. *ACM SIGMETRICS Performance Evaluation Review, 43 (1),* 151–164. https://doi.org/10.1145/2796314.2745875
36. Liu, X., Chen, T., Qian, F., Guo, Z., Lin, F. X., Wang, X. et al. (2017). Characterizing Smartwatch Usage in the Wild. *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services,* 385–398. https://doi.org/10.1145/3081333.3081351
37. Abdelmoneem, R. M., Benslimane, A., Shaaban, E. (2020). Mobility-aware task scheduling in cloud-Fog IoT-based healthcare architectures. *Computer Networks, 179,* 107348. https://doi.org/10.1016/j.comnet.2020.107348
38. Dubey, H., Yang, J., Constant, N., Amiri, A. M., Yang, Q., Makodiya, K. (2015). Fog Data: Enhancing Telehealth Big Data Through Fog Computing. *Proceedings of the ASE BigData & SocialInformatics 2015,* 1–6. https://doi.org/10.1145/2818869.2818889
39. Dubey, H., Monteiro, A., Mahler, L., Yang, Q., Mankodiya, K. (2016). FIT: A Fog Computing Device for Speech TeleTreatments. *2016 IEEE International Conference on Smart Computing (SMARTCOMP).* https://doi.org/10.13140/RG.2.1.1023.8328
40. Rahmani, A. M., Gia, T. N., Negash, B., Anzanpour, A., Azimi, I., Jiang, M. et al. (2018). Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: A fog computing approach. *Future Generation Computer Systems, 78,* 641–658. https://doi.org/10.1016/j.future.2017.02.014
41. Gia, T. N., Jiang, M., Rahmani, A.-M., Westerlund, T., Liljeberg, P., Tenhunen, H. (2015). Fog Computing in Healthcare Internet of Things: A Case Study on ECG Feature Extraction. *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing.* Liverpool: IEEE, 356–363. https://doi.org/10.1109/cit/iucc/dasc/picom.2015.51
42. AMD Geode LX Processors Data Book (2009). *AMD.* Available at: https://www.amd.com/content/dam/amd/en/documents/archived-tech-docs/datasheets/33234H_LX_databook.pdf
43. Gomez, K., Rasheed, T., Riggio, R., Miorandi, D., Sengul, C., Bayer, N. (2013). Achilles and the tortoise: Power consumption in IEEE 802.11n and IEEE 802.11g networks. *2013 IEEE Online Conference on Green Communications (OnlineGreenComm). Piscataway: IEEE,* 20–26. https://doi.org/10.1109/onlinegreencom.2013.6731023
44. Bulić, P., Kojek, G., Biasizzo, A. (2019). Data Transmission Efficiency in Bluetooth Low Energy Versions. *Sensors, 19 (17),* 3746. https://doi.org/10.3390/s19173746

✉***Ivan Shevtsov,*** *PhD Student, Department of Electronic Computers, Kharkiv National University of Radio Electronics, Kharkiv, Ukraine, ORCID: https://orcid.org/0000-0003-0597-1589, e-mail: ivan.shevtsov@nure.ua*

Tetiana Fesenko, *Doctor of Technical Sciences, Professor, Department of Electronic Computers, Kharkiv National University of Radio Electronics, Kharkiv, Ukraine, ORCID: https://orcid.org/0000-0001-9636-9598*

✉*Corresponding author*