Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 151 (2019) 287–294

Procedia Computer Science

www.elsevier.com/locate/procedia

The 10th International Conference on Ambient Systems, Networks and Technologies (ANT)
April 29 – May 2, 2019, Leuven, Belgium

Performance Testing for VoIP Emergency Services: a Case Study of the EMYNOS Platform

Budankailu Sameer Kumar Subudhi[a], Faruk Catal[a], Nikolay Tcholtchev[a]*, Kin Tsun Chiu[a] and Yacine Rebahi[a]

[a] *Fraunhofer FOKUS, Kaiserin-Augusta-Allee 31, Berlin, 10589, Germany*

Abstract

VoIP emergency communications is a promising approach to improving the safety of citizens worldwide. The transition required in this scope includes substituting the legacy PSTN/SS7 based emergency call system by Next Generation IP based components for call establishment and control. Thereby, SIP is used as a session control protocol and RTP as the means to transfer emergency data between the caller and the corresponding Public Safety Access Point (PSAP). The emergency data is not only restricted to voice communication but can cover a rich variety of data, which can be acquired by different means (including the end user devices) and transmitted over IP. This includes video, geo-positioning data, voice, Real Time Text, and sensor data in line with emerging IoT architectures and approaches. A vital aspect in this scope is given by the performance of the underlying network, including its capability to establish calls in emergency situations and to transfer the data required for serving the situation. Therefore, in this paper we evaluate the computational performance of the most recent VoIP emergency system implementation, which was developed by the H2020-EMYNOS project as a realization of the EENA NG112 Long Term Definition (LTD) vision. We perform a series of trials and evaluate the performance of the EMYNOS system in a multi-party lab environment established during the project. We evaluate the time needed to perform basic emergency call operations over IP, whilst in parallel generating Internet type of background traffic. Correspondingly, we worked out a methodology and implemented it in our testbed, which are both presented in the current paper. The obtained numerical results lead to the conclusion that SIP based emergency services stand a good chance to replace legacy systems when it comes to their performance.

Keywords: *performance testing; SIP; VoIP; NG112; PSAP; NGN; background traffic*

* Corresponding author. Tel.: +49 30 3463-7175
E-mail address: nikolay.tcholtchev@fokus.fraunhofer.de

1877-0509 © 2019 The Authors. Published by Elsevier B.V.

10.1016/j.procs.2019.04.041

1. Introduction

The nExt generation eMergencY commuNicatiOnS (EMYNOS) project [1] is a European-funded project aiming at addressing the current emergency systems limitations, particularly in terms of media and location. This project, that ended February 2018, was discussed in details in our paper [2] and produced a platform for enabling European citizens to make IP based emergency calls (to police, ambulance and fire brigade). Indeed, the output of EMYNOS was a Next Generation emergency call full chain, starting from the caller, going through the network (in particular, the ESInet [3]), and ending at the emergency call taker client. Various functionalities such as emergency calls identification, caller/device location configuration, routing to the appropriate emergency call center, location information visualization, sensor data transmission, and protection against false calls were addressed. The EMYNOS platform intersects the NG112 architecture, described in the EENA NG112 LTD document [3], and implements the related functionalities according to the project consortium needs and requirements.

In order to evaluate the EMYNOS framework, a number of experiments were conducted on various sites. This includes the overall development testbed established at the TEIC's premises in Crete, as well as the local Fraunhofer FOKUS playground in Berlin [4]. The undertaken testing activities were initially discussed in [2]. It is worth to mention that in the test cases performed in [2], background traffic was not taken into account as the testbed was not fully operational. In the current paper we present the final results of the performance testing procedures within the EMYNOS project, including various advanced scenarios and the evaluation with supplementary background traffic. The traffic was generated based on open source tools, which are widely used in the community, and a corresponding methodology and belonging test components were developed and scripted – once for the generation and measurement of the SIP traffic and once for the background data. Thereby, the innovation of this paper is given by exemplifying on how NGN based emergency platforms can be tested regarding their performance and their operational limits. Since the EMYNOS constitutes a pioneer in this type of platforms, the VoIP features of NGN 112 are systematically examined in an innovative setup which is rarely available to the public (especially as open source).

The structure of this paper is as follows: section II discusses the current state of the art. Section III presents the overview of the EMYNOS platform. We describe the testbed in section IV. In section V we discuss about the setup followed by the testing methodology and numerical results in section VI. Section VII concludes the paper.

2. Related Work

The utilization of VoIP and SIP for emergency communication has been a topic for intensive research during the past decades with [6], [7], [8], [9] and [10] being some of the research works in very recent years. Besides a European mandate for emergency communication M.493 [11] has been established by ETSI and the European Commission, in order to regulate the setup of VoIP emergency services' architectures according to European law and jurisdiction. It is also worth mentioning that NG911 (perceived as NG112 in Europe) was initially pushed for in the USA thereby establishing standards for facilitating the adoption of this promising technology [12]. Furthermore, a number of European projects have investigated emergency call architectures over IP with [13], [14], [15] and [16] being some of the notable examples in the area. It is also important mentioning that the topic of eCall [18], i.e. the automated triggering of emergency calls, has led to various standards utilizing the traditional SS7 [17] control stack. Especially in the automotive domain, eCall is close to deployment being pushed by corresponding European regulations.

The testing work related to Next Generation emergency services and following the EENA NG112 LTD vision is very scarce. Two main activities were undertaken in 2016 and 2017 under the umbrella of the ETSI NG112 plugtests [5]. These activities were focusing more on interoperability. To our knowledge, performance testing in the context of Next Generation emergency services was presented first by the authors of this paper in [2] where no background traffic was considered.

3. EMYNOS Platform

The EMYNOS platform has been developed to enable, in particular, smarter communication services in case of emergencies. The involved actors can be classified into two categories; namely the Caller and the Callee. Fig. 1 provides an overview of the complete platform. More information on the actors and the platform are mentioned below.

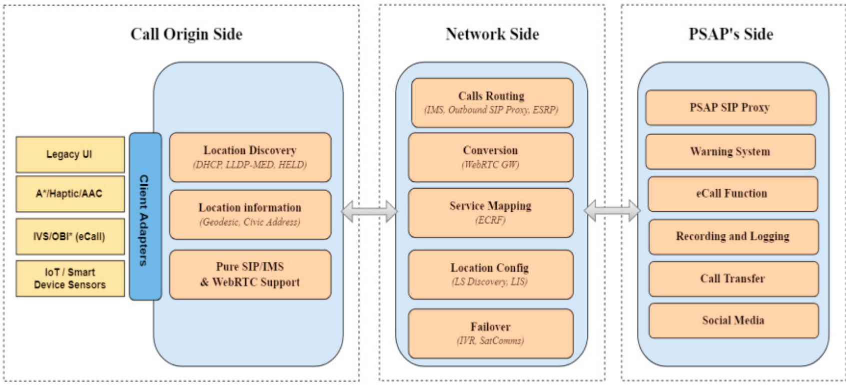

Fig. 1: EMYNOS high-level Architecture

The Callee, in the EMYNOS framework, can be any agency that provides assistance services to handle any emergency situation the Caller is in. Callees may include generic Public Safety Answering Point (PSAP) call takers or even a purpose specific agency such as police, ambulance or firefighting services. The Caller in EMYNOS can be any person placing an emergency call or even a device that implements a SIP or WebRTC enabled client. The generated calls can be initiated either automatically or manually. Devices such as smartwatches, haptic-sensor or any other (health-)sensor (e.g. smart t-shirt) based devices can trigger an automatic emergency call. A person can also initiate an emergency call manually. All these calls are generated from a Voice over Internet Protocol (VoIP) client that internally makes use of the Session Initiation Protocol (SIP) and Web Real-Time Communication (WebRTC) protocols. Sometimes, these caller devices may also send their location information to assist the platform in smartly resolving the closest relevant emergency response service. The payload information for resolving the location-based service in the calls are supported on the above devices using Dynamic Host Configuration Protocol (DHCP), Link Layer Discovery Protocol – Media Endpoint Discovery (LLDP-MED) or HTTP – Enabled Location Delivery (HELD) protocols. The network components of the EMYNOS framework consist of either an outbound SIP proxy or an IP Multimedia Subsystem (IMS) that interacts with a HELD based Location Server to determine the location-based service, then to an Emergency Services Routing Proxy (ESRP) and an Emergency Call Routing Function (ECRF) that help in final routing of the call to the appropriate emergency call center, based on the type of the emergency call and the caller location.

4. Testbed

A testbed implementing all the intricate details stated above has been setup for development, evaluation and demonstration purposes at the Technological Educational Institute of Crete, Greece (TEIC) premises on which all the project partners have been working primarily. The same setup (see Fig. 2) has also been replicated locally at the premises of Fraunhofer Institute for Open Communication Systems (FOKUS), on which the performance tests were run and the results are presented here.

As depicted in Fig. 2, the testbed components consist of a SIP Proxy (Kamailio based), an ESRP (Kamailio based), couple of iPerf instances (Client & Server nodes generating TCP and UDP traffic in parallel) and a couple of SIPp instances (Caller & Callee modes). The iPerf instances are used to generate and mimic the background traffic in the network in a controlled manner. SIPp is used to mimic the emergency callers and callees. The Asterisk - acting as

PBX (Private Branch Exchange) - is required to distribute and delegate the high volume of incoming calls during the emergencies among the various agents available at the PSAP. And in certain cases where all the callees are busy, the PBX component also doubles as an Interactive Voice Response (IVR) system to respond the callers with a pre-recorded message.

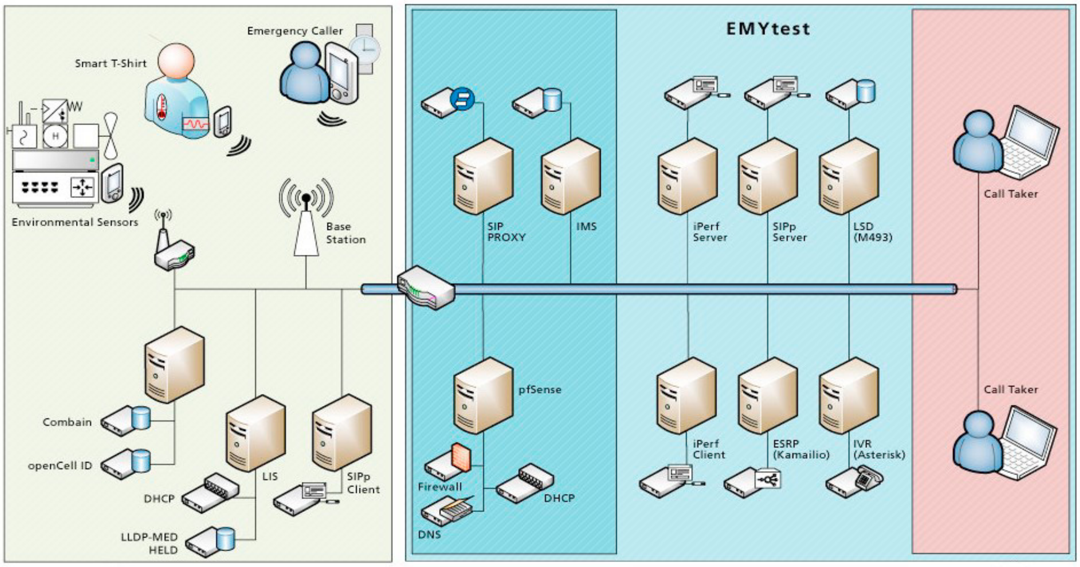

Fig. 2: EMYNOS Testbed in Fraunhofer FOKUS Premises

5. Experimental Setup

The envisioned test was aimed at determining the performance of the primary components of the EMYNOS platform (including the SIP proxy and eventually the ESRP) and its ability to handle the traffic under distress, as would be the case whenever an emergency scenario occurs. To enable the testing, other supportive tools were used, namely SIPp and iPerf. Two SIPp instances were used to emulate the emergency calls in the roles of the emergency caller and the emergency call taker. Also, to determine the bandwidth of the network and emulate the background traffic, two instances of iPerf were used. All the test components in the testbed are setup on separate Virtual Machine (VM) instances to run separately and independent of each other. The VMs host Ubuntu Linux 16.04 on a virtualized environment consisting of Intel Xeon CPU E5420 @ 2.50 GHz, 2.0 GB RAM and 9.4 GB HDD. A major part of the performance testing procedure has been automated using shell scripts. The shell scripts have been developed to be dynamic and parameterized, and use configuration files specific to each available testbed.

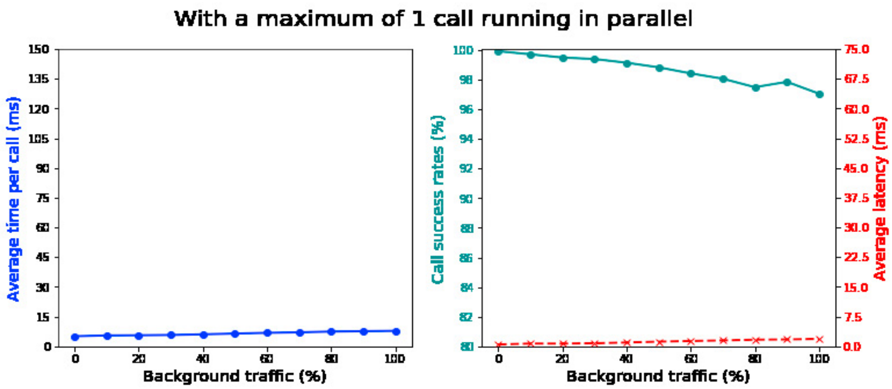

Fig. 3: SIP Call Establishment with maximum 1 call in parallel

This setup has also been used to evaluate various tools under consideration. The tools used for the tests, apart from the various components constituting the EMYNOS platform, can be fit into two categories: 1) The Actor Emulators - as specified earlier, we use SIPp to emulate the actors in this case, namely the Caller and the Callee. 2) Network traffic Simulators - again as previously mentioned, we use iPerf to simulate and saturate the network bandwidth with background traffic in a controlled manner.

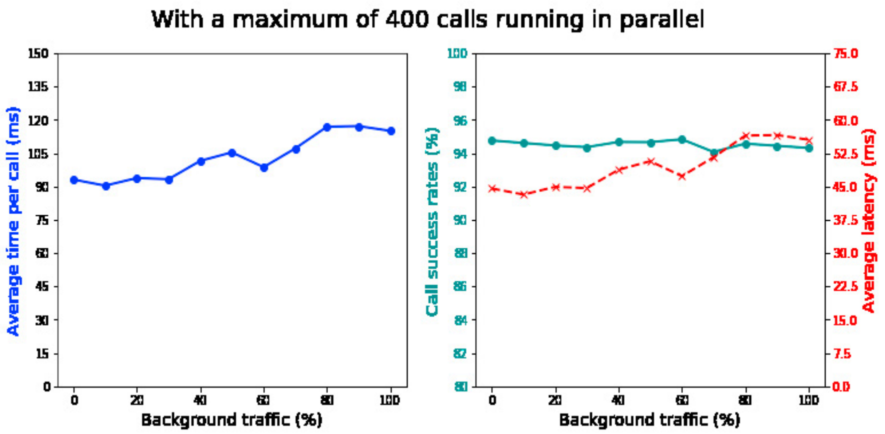

Fig. 4: SIP Call Establishment with a maximum 400 calls in parallel

6. Methodology & Numerical Results

SIPp, the emulator for the actors involved in an emergency call, lets us define custom call scenarios to mimic various user actions in an XML file called the scenario file. These scenario files can also dynamically accept various values from an injection file. We use the injection file methodology to inject the caller names, authentication and various other details dynamically from a list that had one thousand callers defined in it.

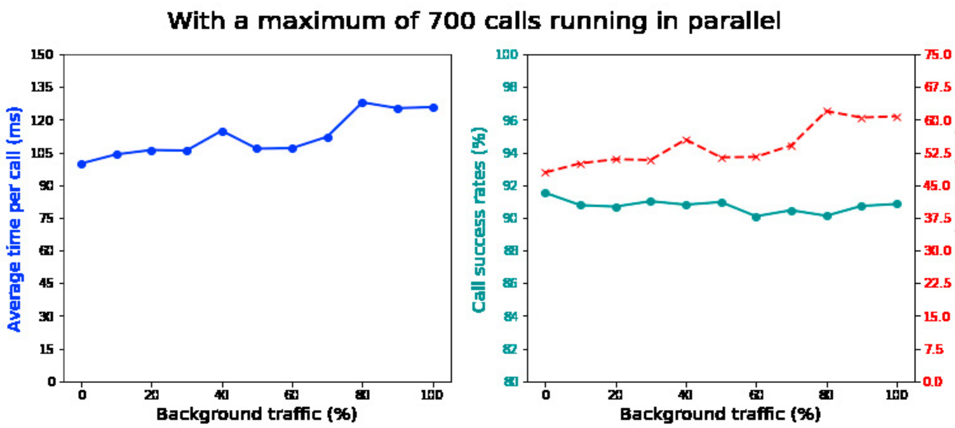

Fig. 5: SIP Call Establishment with a maximum of 700 calls in parallel

With *Average Time per call*, *Average Latency* and *Call Success Rates* as the required metrics in mind, we designed an end-2-end call scenario. In the custom scenario we fabricated, the Caller places an emergency call to the Callee over the EMYNOS platform. As soon as the Callee responds to the call successfully, the call is ended by the Caller. Since all the calls take place over the Internet supported infrastructure, the communication protocol put to use is SIP, as was mentioned earlier. The next paragraph explains the scenario in technical terms.

SIPp emulates the Caller scenario as follows. It begins with sending out a *REGISTER* request to the ESRP. In case the request requires an authentication, the same *REGISTER* request is resent now along with the authentication details. Once the Caller is successfully registered with the ESRP, it sends out an INVITE message to the Callee with priority set to emergency, geo-location and other necessary accompanying data payload in the *Session Description Protocol (SDP)*. The call priority is set using the Priority field in the *INVITE* message as *"Priority: emergency"*. The Caller then expects a *100 TRYING* followed by a *180 RING* and a final *200 OK* that confirms that the Callee has answered the call. The Caller then sends out an *ACK* message immediately followed by a *BYE* to terminate the call. Once the Caller receives the final *200 OK*, corresponding to the *BYE*, from the Callee the emergency call scenario terminates. Contrary to the Caller, the Callee emulation takes place slightly differently in SIPp. The Callee registers itself with the ESRP as a service just once throughout the complete testing procedure. This is required as the Callee needs to be available online all the time to accept an emergency call. Once the Callee is registered to the ESRP,

SIPp now runs the Callee response scenario. The Callee response scenario starts with expecting an *INVITE* request. As soon as it receives an *INVITE* request, it responds with a *180 RINGING* immediately followed by a *200 OK*, which informs the Caller that the Callee has now accepted the call. Then, corresponding to the Caller scenario definition, the Callee now expects to receive an *ACK* message immediately followed by a *BYE*. As soon as the Callee receives a *BYE* message, it acknowledges the BYE message with a *200 OK* message and finally terminates the call from its perspective.

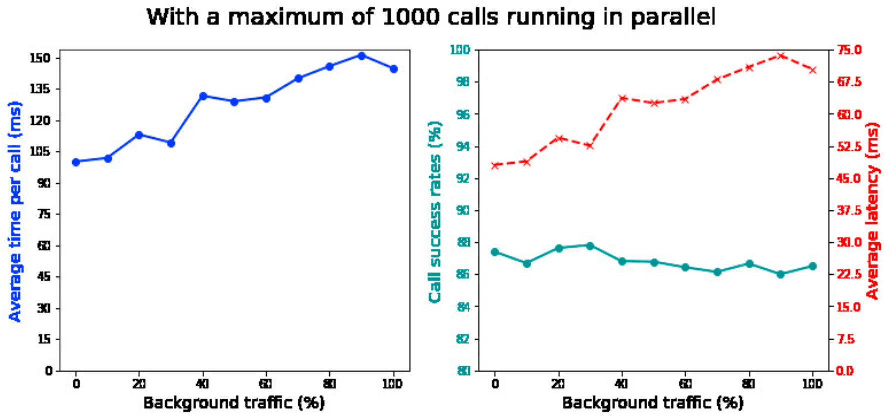

Fig. 6: SIP Call Establishment with a maximum of 1000 calls in parallel

With the understanding of how the Caller and the Callee response scenarios have been implemented, let us now explain what timing metrics are captured and how are they used to derive other details. In the Caller scenario definition, we start the timer when the first *INVITE* message is sent out to the Callee. This timer ends once the call is successfully established and the Caller ends the call with the sending of a *BYE* message. Let us refer this as *ResTimeCaller*. A similar timer is captured on the Callee response scenario. This timer starts once the Callee receives an *INVITE* request from the Caller and terminates with the sending of *200 OK* message which is sent out in response to the *INVITE*. Let us refer this as *ResTimeCallee*. The above call scenario was run multiple times to simulate various kinds of stress level scenarios. Thereby, the goal is to push the available emergency communication platform to the extreme and show how its limits can be examined. The two parameters that were controlled and varied to create the scenarios are:

<u>Maximum number of calls in parallel:</u> We ran the tests by controlling the maximum number of calls that are allowed to be established at any point in time during the test cycle[†]. We ran the tests by executing the calls first sequentially; the subsequent runs allowed 100 calls to exist simultaneously. With every run the number of calls allowed to run in

† The possible statistical variance due to the Internet segment between the end users' devices and the EMYtest is taken care of by starting a large number of calls within which trial and obtaining the average value of the established performance indicators.

parallel was increased by 100 until the cap reached 1000 i.e. the values used for the tests were 1, 100, 200, 300, 400...1000.

Level of background traffic: For every setting above, the communication channel was saturated with background traffic to observe its effect on the captured metrics. The maximum bandwidth supported by the SUT (System under Test) network was determined by using the *iPerf* tool. Once the maximum supported bandwidth was determined, the test scenarios were increased by 10% in every test run starting with absolutely no background traffic.

By combining the two factors explained above, we executed exactly 121 test runs, one for each possible scenario. Each test run was executed with 10000 calls. Thus, for the complete test set that we ran simulated exactly 12.1 million calls. For every scenario that we ran, the average of the *ResTimeCaller* and the average of the *ResTimeCallee* was calculated but only for those calls that completed executing successfully. Let us refer these values as *avgResTimeCaller* and *avgResTimeCallee* respectively.

In this paper, we present results from a few selected cases. The results are presented as graphical plots. The graphs plot all the three metrics mentioned earlier (i.e. *Average Time per call*, *Average Latency* and *Call Success Rates*). We define these metrics as:

Each result set presented here is segregated by the maximum number of calls that were allowed to simulate in parallel. All the three metrics are plot against the background traffic. Every image consists of two graphs. The first graph plots the *Average Time Taken per Call* in milliseconds. The second graph plots the *Call Success Rate* in percentage on the left and the *Average Latency* in milliseconds on the right. The *Call Success Rate* is plotted with the solid line and the *Average Latency* with the dashed line. The results with the corresponding graphs are plotted in Fig. 3 – 6 respectively.

The obtained results illustrate the limits of the open source EMYtest platform for emergency communications within the described setup. Thereby, our goal was mainly to demonstrate and show a methodology regarding how such performance parameters can be measured for innovative scenarios in a VoIP emergency context. In a production grade environment, a series of improvements and hardware upgrades would lead to improved performance within the provided communication requirements.

7. Summary & Outlook

The current paper presents our activities related to the performance testing of VoIP emergency services. Given the advantages provided by NG112 IP based emergency communications, it is a lucrative step to initiate a transition from legacy PSTN architectures to IP based emergency architectures. These advantages include the potential integration of different types of communication such as voice, video, and real time text for providing advanced options for handling an emergency situation. Furthermore, it is possible to integrate data from social networks as well as sensor data, e.g. environmental sensors or health sensors monitoring the human body. Despite all these advantages, the introduction of NG112 comes with a couple of pitfalls. Many experts doubt the actual performance (in terms of computational time and communications) of an IP based emergency system. Hence, there is a need for methods, tools and testbeds that will support the evaluation of emergency services over IP. Therefore, in the current paper we present the EMYNOS testbed for emulating emergency services involving the whole variety of components (DHCP servers, LOST and HELD servers, sensor network integration, ESRP, PSAP, etc.) according to various standards such as the European M.493 and the US Next Generation 911. Based on a described experimental setup, a large amount of measurements were conducted, which are correspondingly presented and show that VoIP emergency services stand a chance to perform reasonably and substitute legacy PSTN systems thereby increasing the richness of provided data and improving the chances to save human life in general.

With respect to future work, we plan the increased integration of sensor devices from the domain of e-health and body networks. Furthermore, the interplay between NG112 and emerging IoT architectures will be investigated and corresponding performance measurements conducted. Thereby, the methodology proposed in this paper will further evolve over time and will give the possibility to explore the characteristics of these emerging networks and make appropriate design decisions.

References

[1]The EMYNOS project, Link. www.emynos.eu, as of date 09.01.2019

[2] Y. Rebahi, K. T. Chiu, N. Tcholtchev, S. Hohberg, E. Pallis, E. Markakis, "Towards a Next Generation 112 Testbed: The EMYNOS ESInet", to appear in the Elsevier International Journal of Critical Infrastructure Protection

[3] Next Generation 112 Long Term Definition, Link: https://de.scribd.com/document/89835743/Eena-Ng112-Ltd-v1-0-Final, as of date 27.02.2019

[4] Fraunhofer FOKUS EMYtest: https://www.fokus.fraunhofer.de/en/sqc/technologies/emytest, as of date 09.01.2019

[5] ETSI NG112 Plugtest #2, Link: http://www.etsi.org/news-events/events/1110-ng112-2, as of date 09.01.2019

[6] M. Manso *et al*., "The Application of Telematics and Smart Devices in Emergencies: Use Cases in Next Generation Emergency Services," *2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI)*, Berlin, 2016, pp. 289-292.

[7] E. Sdongos, A. Bolovinou, M. Tsogas, A. Amditis, B. Guerra and M. Manso, "Next generation automated emergency calls - Specifying next generation ecall & sensor-enabled emergency services," *2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, Las Vegas, NV, 2017, pp. 1-6.

[8] E. K. Markakis, A. Lykourgiotis, I. Politis, A. Dagiuklas, Y. Rebahi and E. Pallis, "EMYNOS: Next Generation Emergency Communication," in *IEEE Communications Magazine*, vol. 55, no. 1, pp. 139-145, January 2017.

[9] Evangelos K. Markakis, Ilias Politis, Asimakis Lykourgiotis, Yacine Rebahi, George Mastorakis, Constandinos X. Mavromoustakis, Evangelos Pallis, "Efficient Next Generation Emergency Communications over Multi-Access Edge Computing", Communications Magazine IEEE, vol. 55, pp. 92-97, 2017, ISSN 0163-6804.

[10] R. Barnes and B. Rosen, "911 for the 21st Century," in *IEEE Spectrum*, vol. 51, no. 4, pp. 58-64, April 2014.

[11] M/493, Standardization Mandate to the European Standards Organizations (ESO) in support of the Location Enhanced Emergency Call Service, Accepted by ETSI at Board#84 on 7 November 2011

[12]NENA Standards & Other Documents, NENA Functional and Interface Standards for Next Generation 9-1-1, online available: https://www.nena.org/page/standards, as of date 09.01.2019

[13] H2020-NEXES NEXt generation Emergency Services, http://nexes.eu/, as of date 09.01.2019

[14] FP7- EmerGent, http://www.fp7-emergent.eu/, as of date 09.01.2019

[15] Help-112, http://eena.org/pages/help-112 , as of date 09.01.2019

[16] Reach-112, http://www.omnitor.se/en/projects/reach-112/, as of date 09.01.2019

[17] ITU-T Recommendation: Q.700 : Introduction to CCITT Signalling System No. 7, online: http://www.itu.int/rec/T-REC-Q.700/en, as of date 09.01.2019

[18] EU Commission, Digital Single Market Policy, eCall: Time saved = lives saved, online: https://ec.europa.eu/digital-single-market/ecall-time-saved-lives-saved, as of date 09.01.2019