JOURNAL OF TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY

SPECIAL ISSUE 2025

Post 63rd FITCE International Congress publication on New Technologies and Services for Cybersecurity Opportunities and Threats

Editor-in-Chief

Adrian Kliks, Poznan University of Technology, Poland

Editorial Advisory Board

Hovik Baghdasaryan, National Polytechnic University of Armenia, Armenia
Naveen Chilamkurti, LaTrobe University, Australia
Luis M. Correia, Instituto Superior Técnico, Universidade de Lisboa, Portugal
Pedro Crespo Bofill, Universidad de Navarra, Spain
Luca De Nardis, DIET Department, University of Rome La Sapienza, Italy
Nikolaos Dimitriou, NCSR "Demokritos" Athens, Greece
Ciprian Dobre, Politechnic University of Bucharest, Romania
Piotr Gawrysiak, Warsaw University of Technology, Poland
Filip Idzikowski, Poznan University of Technology, Poland
Andrzej Jajszczyk, AGH University of Science and Technology, Poland
Zbigniew Jaroszewicz, National Institute of Telecommunications, Poland
Albert Levi, Sabanci University, Turkey
Marian Marciniak, National Institute of Telecommunications, Poland
George Mastorakis, Technological Educational Institute of Crete, Greece
Constandinos Mavromoustakis, University of Nicosia, Cyprus
Takumi Miyoshi, Shibaura Institute of Technology, Japan
Klaus Mößner, Technische Universität Chemnitz, Germany
Imran Muhammad, King Saud University, Saudi Arabia
Mjumo Mzyece, University of the Witwatersrand, South Africa
Daniel Negru, University of Bordeaux, France
Jordi Perez-Romero, UPC, Spain
Michał Pióro, Warsaw University of Technology, Poland
Konstantinos Psannis, University of Macedonia, Greece
Salvatore Signorello, University of Lisboa, Portugal
Adam Wolisz, Technische Universität Berlin, Germany
Tadeusz A. Wysocki, Univeristy of Nebrasca, USA

Editorial Team

Content Editor: **Robert Magdziak**
Managing Editor: **Ewa Kapuściarek**

eISSN 1899-8852

© Copyright by National Institute of Telecommunications, Poland 2025

Introduction to the Special Issue

We are pleased to present this special issue of the *Journal of Telecommunications and Information Technology*, which features selected scientific papers from the **63rd FITCE Congress** (Federation of Telecommunications Engineers of the European Community), held in Kraków, Poland, in September 2024. The central theme of the Congress was *"New Technologies and Services – Opportunities and Threats"*, explored through four thematic sessions focused on cybersecurity, technologies, and their applications. Additional special sessions were organized for young Ph.D. students and representatives of the telecommunications industry.

The articles presented in this issue address key challenges related to ensuring communication security and privacy, the application of machine learning and artificial intelligence methods, and the integration of best industry practices – topics of critical importance in today's digital landscape. This special issue showcases research efforts aimed at solving technical problems through original systemic approaches that enhance the state-of-the-art in communication technologies.

The included articles address the following topics:

- The role of cybersecurity certification within the upcoming EUCC framework, with practical approaches to laboratory accreditation and penetration testing methodologies.
- Advanced machine learning-based detection of malware-generated domains (DGA) using both classical and neural models.
- Application of differential privacy in biomedical signal analysis, ensuring the secure and ethical use of patient data.
- Optimization of military ad hoc networks to reduce detectability via unmanned vehicles.
- Cyber and operational challenges associated with the shutdown of traditional PSTN systems and the transition to all-IP networks.
- A techno-economic analysis of cybersecurity of IoT and OT, including implications for policy and market regulation.
- Evaluation of resource allocation mechanisms in 6G V2X networks, with a focus on privacy and system performance trade-offs.

Each paper included in this issue has undergone a peer review process and was selected based on its originality, relevance, and potential impact on the telecommunications field.

As Guest Editors, it has been a privilege to oversee the development of this special issue. We were impressed by the breadth and quality of the submissions, which reflect not only the dynamic and evolving nature of the field but also the collaborative spirit of the FITCE community. We extend our sincere thanks to all the authors for their outstanding contributions and to the anonymous reviewers for their careful and constructive evaluations.

We also wish to express our gratitude to the *JTIT* Editorial Board for their support and to the FITCE 2024 Organizing Committee for hosting an exceptional and inspiring Congress in Kraków.

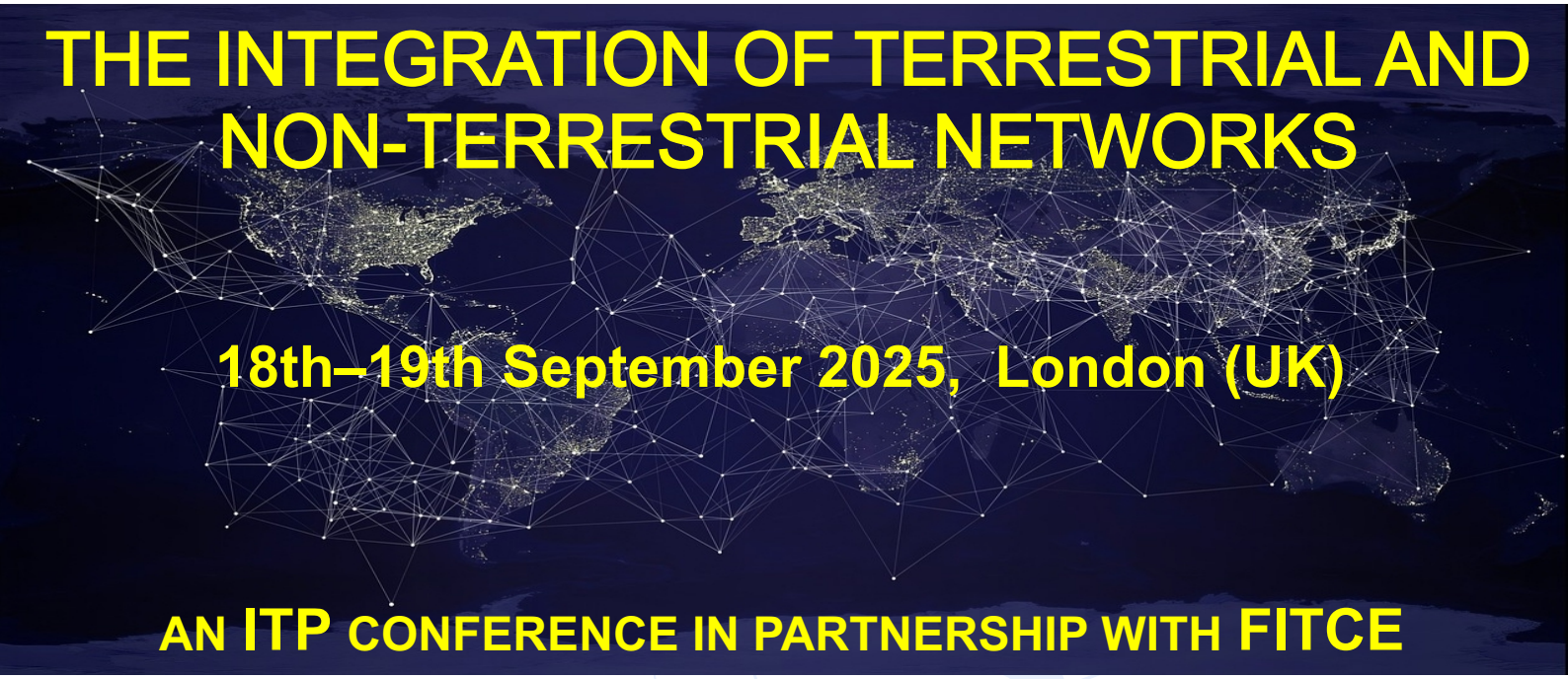
We hope this special issue will serve as a valuable reference for readers and inspire further research and innovation at the intersection of telecommunications, cybersecurity, and emerging digital technologies.

Guest Editors:

George Agapiou, Andy Valdar, and Piotr Zwierzykowski

64th FITCE CONGRESS

Federation of Telecommunications Engineers of the European Union (FITCE, based in Bruxelles), in partnership with the Institute of Telecommunications Professionals (ITP) recognises the growing interest in space-based communications and the role it can play in delivering an optimised heterogeneous network with wide coverage and high availability, invite you to a conference:



SIT together with the organizers invites you to the conference

The Polish Association of Telecommunication Engineers SIT is forum for the exchange of new results among engineers, scientists and researchers on advances in telecommunication and other related areas. SIT has been a member of the FITCE since 2002.

For more information scan QR code or visit:

https://www.theitp.org/events/itp-conference-the-integration-of-terrestrial-and-non-terrestrial-networks/

k-anonymity in Resource Allocation for Vehicle-to-Everything (V2X) Systems

Andres Vejar, Faysal Marzuk, and Piotr Chołda

AGH University of Kraków, Kraków, Poland

https://doi.org/10.26636/jtit.2025.FITCE2024.1998

Abstract — Sixth generation (6G) vehicle-to-everything (V2X) systems face numerous security threats, including Sybil and denial-of-service (DoS) cyber-attacks. To provide a secure exchange of data and protect users' identities in 6G V2X communication systems, anonymization techniques – such as k-anonymity – can be used. In this work, we study centralized vs. k-anonymity based resource allocation methods in a vehicular edge computing (VEC) network. Allocation decisions for vehicular networks are classically posed as a centralized optimization task. Therefore, an information flow is transmitted from the vehicles to the communication premises. In addition to a resource allocation decision, vehicle information is not required. We analyze the centralized allocation versus k-anonymous allocation models. To show a potential deterioration introduced by anonymity, we quantify the gap in the optimal goal in two cases: based on resource allocation and with aim at energy reduction. Our numerical results indicate that energy consumption rises by 1% in smaller scenarios and 23% in medium scenarios, whereas it decreases by 14% in larger scenarios.

Keywords — k-anonymity, privacy-enhancing technologies, resource allocation in 6G, vehicle-to-everything (V2X) systems

1. Introduction

Sixth generation (6G) networks are expected to facilitate and enhance the services of intelligent transportation systems (ITS) by integrating artificial intelligence (AI) techniques with machine learning (ML) algorithms [1]. The vehicle-to-everything (V2X) system, which is an application of ITS, enables the exchange of information between vehicles and their surroundings [2]. Vehicles can communicate through vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications, such as the roadside unit (RSU), as shown in Fig. 1. The newly proposed 6G V2X communication systems can easily be targeted by different security attacks due to their high mobility, highly dynamic topology, and variety of communications [3].

The deployment of AI techniques in the design of vehicular edge computing (VEC) networks has limitations due to robust security mechanisms, considerations of privacy and ethics, as well as new security developments [1]. The collection and processing of data in VEC systems require the protection of user privacy with privacy-enhancing technologies (PET), including differential privacy and data anonymization methods, to reduce the risk of re-identification and unauthorized monitoring [1].

Several applications of PETs involve k-anonymity [4] and its variations [5], [6]. The privacy and efficiency requirements in vehicular networks can be addressed using k-anonymity. To achieve these requirements, k-anonymity with differential privacy can be combined with transactional blockchain registration [7].

A framework for the sharing of private data within ad hoc vehicular networks (VANET) is introduced using federated learning (FL) and local differential privacy [8]. This approach guarantees protection against inference and gradient leakage attacks while providing higher efficiency than conventional FL-based methods. A local differential privacy technique is used to provide a privacy preservation solution for VANET by excluding the need for a third party to anonymize critical information [9]. The disclosure of sensitive data, such as vehicle positions in location services, is considered a potential threat to the privacy of users [10]. The k-anonymity method is used to maintain location privacy in edge computing (EC) [11], and to preserve location privacy on the Internet of Vehicles (IoV) [12].

Zero-trust architectures that provide privacy by design need to be privileged to provide essential data security and privacy preservation requirements for the 6G V2X allocation process. Due to the various V2X applications, such as V2P and V2V communications, the design of a secure and private management system is a critical concern [13].

Ensuring secure data exchanges requires trusted management in the allocation process. To address the challenge of designing a secure 6G V2X communication system with VEC services, anonymization techniques can be used to protect the identity of users by reducing specific vehicle information. That leads to a reduction of the surface of attack in the V2X infrastructure.

If the resource allocation system is compromised by malicious agents, the identification of each vehicle is available to the

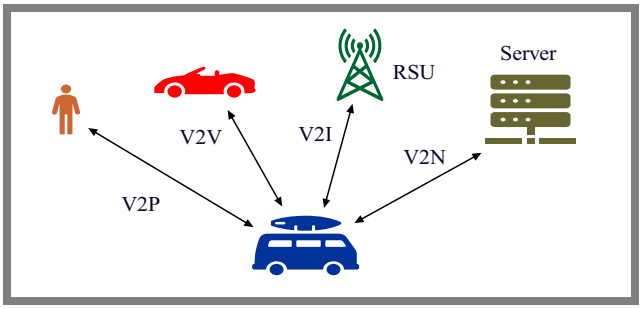


Fig. 1. Types of V2X communications.

This work is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) License.
For more information, see https://creativecommons.org/licenses/by/4.0/

attacker. This information can be used to escalate the attacks to other elements of the 6G V2X system, notably V2V and V2P. In this work, we study the effect of incorporating k-anonymity into the 6G V2X allocation system.

2. System Description

Table 1 summarizes the mathematical notation used to describe the system under study.

We consider a 6G V2X communication system that includes sets of vehicles and RSUs, as shown in Fig. 2. RSUs extend the computation and communication capabilities to vehicles by being deployed closer to end users. In our infrastructure of the system under study, vehicles need to send their data to RSUs for processing (offloading option).

We investigate a scenario consisting of a set of RSUs ($i = 1, 2, \ldots, I$) and a set of vehicles ($j = 1, 2, \ldots, J$). Each RSU i has several available resource blocks (RBs) per time interval, denoted by M_i. Each vehicle j, if it is associated with RSU i, will require several RBs to upload its data, indicated by $R_{i,j}$. The required number of RBs depends on the signal-to interference, noise ratio (SINR) values, and the uplink data rates. We need to determine the optimal assignments between RSUs and vehicles in order to decide whether to turn on or off the RSU. Our objective is to reduce the energy consumption and the number of active RSUs depending on the number of RBs required by each vehicle and subject to uplink bandwidth and uplink time constraints illustrated by SINR and inter-cell interference (ICI). We calculate SINR values for the uplink of

Tab. 1. Mathematical notations used throughout the paper.

Symbol	Meaning
I	Set of RSUs
J	Set of vehicles
P_j	Transmission power of vehicle j
D_j	Communication demand of vehicle j
Φ_j	Computation demand of vehicle j
M_i	Available uplink RBs per time slot for RSU i
F_i	Maximum available computation capacity of RSU i
$\mathcal{L}_j$	Maximum allowed latency for vehicle j
$\mathrm{R_{i,j}}$	Required RBs per time slot to send data
$\gamma_{i,j}$	SINR value for vehicle j and RSU i
$\mathrm{H}_{i,j}$	Threshold value for $\gamma_{i,j}$
$\mathrm{U}_{i,j}$	Required uplink data rate of vehicle j
$\mathcal{U}_{i,j}$	Link capacity between vehicle j and RSU i
ψ_i^J	Energy coefficient of RSU server's chip architecture
x_i	Decision variable to turn on/off the RSU i
$y_{i,j}$	Decision variable indicating whether vehicle j is associated with RSU i or not

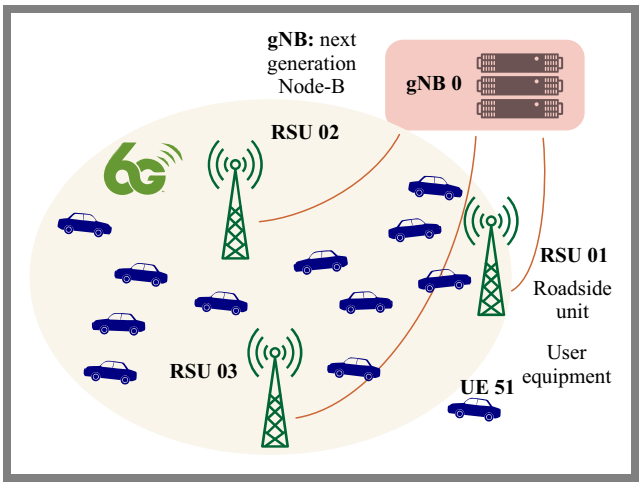


Fig. 2. An example of a V2X communication system.

data from the ICI aggregate uplink, the coverage of the RSU, and the distance between the vehicle and the interference RSU [14].

After calculating the SINR values for each vehicle, we determine the RB's data rates depending on different modulation orders, SINR ranges, and efficiencies from the mapping table given in [15]. This mapping is used to determine the number of required RBs where an RB per time interval of 0.25 ms consists of 12 sub-carriers of 60 kHz spacing and each sub-carrier consists of 14 OFDM symbols. The number of RBs $R_{i,j}$ required by each vehicle to process its data is calculated as $R_{i,j} = U_{n,v} \times (12 \times 14 \times \text{efficiency})^{-1}$.

Considering I RSUs with a number of available uplink RBs (M_i) and J vehicles with a number of required RBs per time slot for uploading the data from vehicle j to the i RSU $R_{i,j}$, we need to determine $y_{i,j}$ which denotes whether the vehicle j is associated with RSU i or not; and x_i which indicates whether to turn the RSU i on or off. We formulate an optimization problem to minimize the energy consumption and the number of active RSUs as:

$$\min\ \omega_1 \sum_{i\in \mathrm{I}} x_i + \omega_2 \sum_{i\in \mathrm{I}} \sum_{j\in \mathrm{J}} \left(\frac{P_j \mathrm{D}_j}{\mathcal{U}_{i,j}} + \frac{\psi_i^J \mathrm{D}_j \Phi_j}{\mathrm{f}_{i,j}^{-2}} \right) y_{i,j}\,, \quad (1)$$

$$\text{s.t.} \quad \sum_{j\in \mathrm{J}} \mathrm{R}_{i,j}\, y_{i,j} \leqslant \mathrm{M}_i, \quad \forall\, i \in \mathrm{I}\,, \quad (2)$$

$$\sum_{j\in \mathrm{J}} \gamma_{i,j}\, y_{i,j} \geqslant \mathrm{H_{i,j}}, \quad \forall\, i \in \mathrm{I}\,, \quad (3)$$

$$\sum_{j\in \mathrm{J}} \mathrm{f}_{i,j}\, y_{i,j} \leqslant \mathrm{F}_i, \quad \forall\, i \in \mathrm{I}\,, \quad (4)$$

$$\sum_{i\in \mathrm{I}} \mathrm{L}_{i,j}\, y_{i,j} \leqslant \mathcal{L}_j, \quad \forall\, j \in \mathrm{J}\,, \quad (5)$$

$$\sum_{i\in \mathrm{I}} y_{i,j} = 1, \quad \forall\, j \in \mathrm{J}\,, \quad (6)$$

$$x_i \geqslant y_{i,j}, \quad \forall\, i \in \mathrm{I}, \quad \forall\, j \in \mathrm{J}\,, \quad (7)$$

$$x_i,\ y_{i,j} \in \{0, 1\}, \quad \forall\, i \in \mathrm{I}, \quad \forall\, j \in \mathrm{J}\,. \quad (8)$$

3. Results

As a case study, we first investigate an allocation scenario consisting of 4 RSUs and 32 vehicles. Table 2 lists the parameter values that are used in the calculations and evaluations, where the values are assumed according to the service requirements for 6G V2X services and to guarantee the QoS requirements of the communications system [16].

Each vehicle, if assigned to an RSU, will upload its computational tasks to be processed. Vehicle information includes:

1) communication demand, indicated by D_j, in the range $10 - 60$ kbits,
2) computation demand, indicated by Φ_j, in the range $100 - 150$ cycles/bit,
3) transmission power, indicated by P_j, in the range $23 - 33$ dBm.

This information D_j, Φ_j, P_j is used to calculate the communication delay, the computation delay, and the energy consumption for centralized allocation [17]. If the allocation system is breached, these details can be exploited to uncover, monitor, and further compromise the privacy of UEs.

Designing systems with enhanced privacy techniques such as k-anonymity or differential privacy can reduce the probability of unwanted or unauthorized tracking and re-identification.

In this work, we use V2V communication to achieve k-anonymity through proximity clusters. We assume that V2V communication is secured in its radius of operation. The triplet D_j, Φ_j, P_j is then distributed in the vehicle proximity cluster, and the aggregate measurement is pooled into its average value.

Each vehicle, by V2I communication, transmits the aggregated triplet values, denoted by $< D_j, \Phi_j, P_j >$ to the RSUs. Similarly, the next generation Node-B (gNB) estimates the SINR values of each vehicle with respect to each RSU. To minimize user information leakage, the SINR values are also aggregated for each proximity cluster and are denoted by $<$ SINR $>$.

The membership in a cluster is verified by the vehicles that share the same value of $< D_j, \Phi_j, P_j >$. The k-private allocation system receives only the aggregated data from vehicles $< D_j, \Phi_j, P_j >$ and the aggregated SINR from gNB, $<$ SINR $>$.

We compare the k-anonymous V2X allocation model presented in Fig. 2 with the centralized allocation model [17]. The scenarios consider an initial density of 126 RSUs/km^2, and a density of vehicles of 1000 vehicles/km^2.

As can be seen in Tab. 3, our numerical results shows that for small and medium scenarios the energy consumption is increased by 1% and 23% respectively while for the large scenario the energy consumption is reduced by 14%.

We note that for the large scenario of 190 vehicles, not all the original constraints are satisfied, allowing for a reduced energy consumption in the k-anonymous version than in the centralized version.

Tab. 2. Parameter values used in the evaluation.

Parameter	Value	Parameter	Value
RSU coverage	200 m	P_j	23 – 33 dBm
M_i	135 RBs	$H_{i,j}$	–7 dB
F_i	20 GHz	$U_{i,j}$	50 – 100 Mbps
$\mathcal{L}_j$	30 ms	D_j	10 – 60 kbit

Tab. 3. k-anonymous versus centralized allocations.

Allocation	No. of RSUs selected/ available	No. of vehicles	Energy
Centralized	2/4	32	0.002432
k-anonymous	2/4	32	0.002459
Centralized	4/16	127	0.005532
k-anonymous	5/16	127	0.006830
Centralized	7/24	190	0.009790
k-anonymous	7/24	190	0.008454

4. Conclusions

In this study, we investigated the incorporation of PETs into the 6G V2X allocation system. Vehicle information was used to calculate communication delay, computation delay, and energy consumption for centralized allocation. We compared centralized resource allocation versus k-anonymous allocation.

Our implementation indicated how variations in optimal allocations are affected when PET is applied to the V2X system. Noting that the k-anonymous technique implemented can be applied to allocation schemes different from the optimal model studied in this work.

Our numerical results illustrated that energy consumption increased by 1% in smaller scenarios and 23% in medium scenarios, while it decreased by 14% in larger scenarios.

Future research will explore enhanced methods, focusing on integrating online allocation through AI models. We plan to explore enhanced methods, focusing on integrating online allocation through AI models. In addition, we plan to evaluate the proposed algorithms in a real world scenario to demonstrate their effectiveness.

Acknowledgments

This research is supported by the National Research Institute, grant number POIR.04.02.00-00-D008/20-01, on “National Laboratory for Advanced 5G Research” (acronym PL-5G) as part of the Measure 4.2 Development of the modern research infrastructure of the science sector 2014–2020 financed by the European Regional Development Fund.

References

[1] M. Humayun *et al.*, "Securing the Internet of Things in Artificial Intelligence Era: A Comprehensive Survey", *IEEE Access*, vol. 12, pp. 25469–25490, 2024 (https://doi.org/10.1109/ACCESS.2024.3365634).

[2] O. Aouedi *et al.*, "A Survey on Intelligent Internet of Things: Applications, Security, Privacy, and Future Directions", *IEEE Communications Surveys & Tutorials*, 2024 (https://doi.org/10.1109/COMST.2024.3430368).

[3] M. AlMarshoud, M.S. Kiraz, and A.H. Al-Bayatti, "Security, Privacy, and Decentralized Trust Management in VANETs: A Review of Current Research and Future Directions", *ACM Computing Surveys*, vol. 56, pp. 1–29, 2024 (https://doi.org/10.1145/3656166).

[4] P. Samarati and L. Sweeney, "Generalizing Data to Provide Anonymity when Disclosing Information (Abstract)", *Proc. of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems. PODS '98*, p. 188, 1998 (https://doi.org/10.1145/275487.275508).

[5] F. Song, T. Ma, Y. Tian, and M. Al-Rodhaan, "A New Method of Privacy Protection: Random k-Anonymous", *IEEE Access*, vol. 7, pp. 75434–75445, 2019 (https://doi.org/10.1109/ACCESS.2019.2919165).

[6] F. Wang, H. Chen, and Y. Zhou. "A Privacy Protection Application of Consumer Personal Information Based on an Improved K-Anonymity Algorithm", *2024 5th International Conference for Emerging Technology (INCET)*, Belgaum, India, 2024 (https://doi.org/10.1109/INCET61516.2024.10593091).

[7] C. Gu, X. Cui, M. Li, and D. Hu. "An Efficient and Privacy-preserving Information Reporting Framework for Traffic Monitoring in Vehicular Networks", *IEEE Transactions on Vehicular Technology*, vol. 72, pp. 7900–7913, 2023 (https://doi.org/10.1109/TVT.2023.3241656).

[8] H. Batool *et al.*, "A Secure and Privacy Preserved Infrastructure for VANETs Based on Federated Learning with Local Differential Privacy", *Information Sciences*, vol. 652, art. no. 119717, 2024 (https://doi.org/10.1016/j.ins.2023.119717).

[9] Z. Iftikhar *et al.*, "Privacy Preservation in the Internet of Vehicles Using Local Differential Privacy and IOTA Ledger", *Cluster Computing*, vol. 26, pp. 3361–3377, 2023 (https://doi.org/10.1007/s10586-023-04002-0).

[10] Z. Qi and W. Chen, "Location Privacy Protection of IoV Based on Blockchain and K-anonymity Technology", *2023 6th International Conference on Electronics Technology (ICET)*, Chengdu, China, 2023 (https://doi.org/10.1109/ICET58434.2023.10211967).

[11] S. Zhang *et al.*, "A Caching-based Dual K-anonymous Location Privacy-preserving Scheme for Edge Computing", *IEEE Internet of Things Journal*, vol. 10, pp. 9768–9781, 2023 (https://doi.org/10.1109/JIOT.2023.3235707).

[12] B. Wang, J. Liu, and L. Dai, "K-anonymity-based Privacy-preserving and Efficient Location-based Services for Internet of Vehicles Withstand Viterbi Attack", *Proc. of International Conference on Image, Vision and Intelligent Systems 2022 (ICIVIS 2022)*, pp. 1016–1028, 2022 (https://doi.org/10.1007/978-981-99-0923-0_101).

[13] M. Georgiades and M.S. Poullas, "Emerging Technologies for V2X Communication and Vehicular Edge Computing in the 6G Era: Challenges and Opportunities for Sustainable IoV", *2023 19th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, Pafos, Cyprus, 2023 (https://doi.org/10.1109/DCOSS-IoT58021.2023.00108).

[14] R.-H. Hwang, F. Marzuk, M. Sikora, P. Chołda, and Y.-D. Lin, "Resource Management in LADNs Supporting 5G V2X Communications", IEEE Access, vol. 11, pp. 63958–63971, 2020 (https://doi.org/10.1109/ACCESS.2023.3288699).

[15] ETSI, "Evolved Universal Terrestrial Radio Access (E-UTRA). Physical Layer Procedures (Technical Specification)", 3GPP TS 36.213 V17.6.0. Release 17, 2024.

[16] K. Sehla, T.M.T. Nguyen, G. Pujolle, and P.B. Velloso, "Resource Allocation Modes in C-V2X: From LTE-V2X to 5G-V2X", *IEEE Internet of Things Journal*, vol. 9, pp. 8291–8314, 2022 (https://doi.org/10.1109/JIOT.2022.3159591).

[17] F. Marzuk, A. Vejar, and P. Chołda, "Optimal Resource Allocation for 6G V2X Communication Systems". *Przegląd Telekomunikacyjny – Wiadomości Telekomunikacyjne*, vol. 97, pp. 350–353, 2024 (https://doi.org/10.15199/59.2024.4.78).

Andres Vejar, Ph.D.
Institute of Telecommunications
https://orcid.org/0000-0002-2041-0387
E-mail: avejar@agh.edu.pl
AGH University of Kraków, Kraków, Poland
https://www.agh.edu.pl

Faysal Marzuk, M.Sc.
Institute of Telecommunications
https://orcid.org/0000-0002-7576-182X
E-mail: faysal.marzuk@agh.edu.pl
AGH University of Kraków, Kraków, Poland
https://www.agh.edu.pl

Piotr Chołda, D.Sc.
Institute of Telecommunications
https://orcid.org/0000-0003-2018-4057
E-mail: piotr.cholda@agh.edu.pl
AGH University of Kraków, Kraków, Poland
https://www.agh.edu.pl

The Proactive Face of Cybersecurity: Certification. Legislation and Market Response from the Perspective of ITSEF

Elżbieta Andrukiewicz and Piotr Krawiec

National Institute of Telecommunications, Warsaw, Poland

https://doi.org/10.26636/jtit.2025.FITCE2024.1984

Abstract — The first European Cybersecurity Certification Scheme according to the Common Criteria (EUCC) specifies a number of additional requirements for Conformity Assessment Bodies (CABs) to be technically competent to provide evaluation and certification services. The NIT Testing Laboratory (ITSEF) has developed a roadmap to meet these requirements and obtain the status of an authorized ITSEF that can provide assessments of ICT products at the "high" assurance level. The roadmap consists of 3 parts: one organizational part concerning the management system and two technical parts concerning evaluations. The paper presents two action points: the innovative approach that NIT ITSEF has implemented regarding the integrated management system in the laboratory in order to achieve optimal cost-benefit ratios and the reliable and verifiable methodology for calculating the attack potential that NIT ITSEF has used to prove that the penetration tests developed and executed on the evaluated software product meet the requirements of AVA_VAN.5. The roadmap will fulfill all the requirements necessary to obtain the status of an authorized ITSEF in the EUCC program.

Keywords — Common Criteria, cybersecurity certification, EUCC, ITSEF, testing laboratory

1. Introduction

Today, European legislators increasingly refer to cybersecurity certification to ensure the proper implementation of many new cyber regulations, such as the Artificial Intelligence Act, the EU Digital Identity Framework, the NIS2 Directive, and the Cyber Resilience Act. The EU requires its member states to rely on cybersecurity certification by providing proactive solutions, often referred to as "compliance" and "presumption of conformity".

The European Cybersecurity Certification Framework, adopted by the European Union in the Cybersecurity Act (CSA) [1], has a twofold objective. First, it aims to help increase trust in ICT products, ICT services, and ICT processes that have been certified under European cybersecurity certification schemes. Second, it should help avoid the proliferation of conflicting or overlapping national cybersecurity certification schemes, thereby reducing costs for businesses operating in the Digital Single Market.

In order to meet the objectives of the European Union, the first European cybersecurity certification scheme is just around the corner. The EU Cybersecurity Certification Scheme on Common Criteria (EUCC) covers the cybersecurity certification of ICT products based on Common Criteria [2] and a Common Methodology for Information Technology Security Evaluation [3] and their corresponding ISO standards, ISO/IEC 15408 and ISO/IEC 18045 respectively.

The EUCC is based on third-party conformity assessments carried out by accredited conformity assessment bodies at two levels: test laboratories providing cybersecurity evaluations and certification bodies issuing certificates based on completed evaluations.

The Common Criteria have proven to be particularly effective over the last two decades in Europe for the certification of integrated circuits and smart cards, thus contributing to increasing the security level of many ICT products, such as electronic signature devices, machine-readable travel documents (passports), bank cards, and digital tachographs.

Poland is one of eight countries in the European Economic Area (EEA) technically and organizationally prepared to evaluate ICT products and issue certificates under the EUCC umbrella. The Polish certification structure consists of an accredited certification body (located at the National Research Institute NASK) issuing cybersecurity certificates and two accredited testing laboratories, the leading and most advanced of which is part of the National Institute of Telecommunications (NIT).

In this article, we present the innovative approach we have taken at the NIT laboratory to become a fully authorized testing entity for the future EUCC scheme. This approach includes the specific implementation of a laboratory management system that seamlessly integrates the requirements of two standards, i.e., ISO/IEC 17025 and ISO/IEC 27001, achieving an optimal cost-benefit ratio.

Furthermore, we propose a reliable and verifiable methodology for calculating the attack potential to prove that the penetration tests developed and executed on the evaluated software product meet the high requirements of AVA_VAN.5 (vulnerability analysis). The proposed methodology fills the gap experienced in software product assessments that require high attack potential due to the lack of any direct references to catalogs containing descriptions of relevant attacks. By using highly systematic methodologies, the NIT laboratory

This work is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) License.
For more information, see https://creativecommons.org/licenses/by/4.0/

achieves the goals of its roadmap, which aims to meet all the requirements necessary to obtain the status of an authorized laboratory in the EUCC program.

2. Related Works

The EU Regulation [4] sets the entry-in-force date of the EUCC on 27 February 2025. As a result, all stakeholders who play their roles in the cybersecurity certification ecosystem have begun final preparations to achieve readiness.

It should be noted that the EUCC scheme follows the pattern of existing schemes used for Common Criteria certificates: certification and evaluation services are provided by different Conformity Assessment Bodies (CABs), called Certification Bodies (CBs), and Testing Laboratories called IT Security Assessment Facilities (ITSEFs). ITSEFs provide cybersecurity evaluation services and CBs issue certificates after the successful completion of ITSEFs' evaluations.

For the assurance level "substantial", no restrictions are provided in [4], except that the CAB must be accredited. The national accreditation body grants accreditation if the CAB meets all the requirements specified in certain international standards. There is no limit to the number of CABs operating in Europe.

However, in order to provide services at the assurance level "high", the certification and evaluation capabilities of the relevant CABs must be additionally confirmed by the National Cybersecurity Certification Authority (NCCA), designated in each Member State. According to [4], separate requirements refer to specific technical domains, that is, "Smart Cards and Similar Devices" and "Hardware Devices with Security Boxes", and ITSEFs demonstrate their capabilities to develop and conduct penetration tests with a specified attack potential. CABs can demonstrate their capabilities in specific application areas and, after successful assessment, the NCCA grants the relevant authorization.

The preparation process in the different existing national schemes will vary depending on the complexity of the conformity assessment body structure and current operational practices in the Member State. An interesting overview of the strategy and its implementation for the German EUCC national structure is given in [5] and [6] and for the Netherlands in [7].

However, a critical factor is that the process of preparing the accreditation requirements, according to [4], has not yet been completed. For example, the state-of-the-art document [8] describing the accreditation requirements for certification bodies contains a reference to the ISO/IEC 19896-3 standard, which is still under development. This standard is needed in the context of the EUCC because it deals with the competence management system to be applied to certifiers.

From a vendor perspective, the certification process is similar to those conducted in the national Common Criteria schemes gathered in the SOG-IS MRA [9]. Certification and evaluation service providers offer workshops, guidelines, and other types of communication to vendors to increase awareness and knowledge. However, this may be only a technical part of the vendor's concerns. Vendors generally express concerns about the additional obligations included in [1]. They point to a shift in responsibility for disclosing and handling vulnerabilities that may be identified in the certified product and other information obligations that are new to them. The fees and penalties in case of inappropriate demonstration of fulfilling the obligations by the vendor, appear to be enormous.

Furthermore, vendors may be negatively affected by the lack of mutual recognition of certificates issued under the EUCC scheme and national certification schemes outside Europe, mainly collected under the global Common Criteria Recognition Arrangement (CCRA) [10]. The lack of mutual recognition may be seen as an additional barrier to the recognition and acceptance of non-EU cybersecurity certificates. It should be noted that around 50% of Common Criteria certificates are issued outside of Europe [11]. A long-term lack of mutual recognition can harm the prospects of the global cybersecurity certification market.

3. Conducted Research

NIT ITSEF began operations in 2019. Since its inception, the ITSEF concept has been based on three principles:

1) ITSEF provides levels of confidentiality and integrity of the target of evaluation equivalent to the assurance level at which the evaluation is conducted,
2) The security requirements for the test laboratory are constructed in exactly the same manner as for the site(s) where the target of evaluation is developed,
3) The ITSEF maintains a constructive interaction with the NIT Cybersecurity Department, which is responsible for cybersecurity research and development (R&D) activities.

The preparation of ITSEF to achieve readiness of ITSEF for EUCC started immediately after the publication of CSA [1]. ITSEF recognized three work packages:

1) Organizational, which covers accreditation requirements for ITSEF,
2) Technical, to support ITSEF authorization; it covers completion of at least one successful evaluation of software products with an attack potential of at least AVA_VAN level 4 of vulnerability assessment class and in accordance with a specified European standard,
3) Technical, to support ITSEF authorization; it covers proving the technical capability of ITSEF to evaluate one or two technical domains, i.e. "Smart Cards and Similar Devices" and "Hardware Devices with Security Boxes".

On the date of submission, the first two work packages have been completed. The third is under development. As such, NIT ITSEF will be the first test laboratory in Poland authorized to conduct ICT product evaluations at the "high" assurance level.

3.1. An Innovative Approach to the ITSEF Management System

Looking at CSA regulations [1], there is a general lack of security requirements to protect the evaluation process, including maintaining the confidentiality and integrity of the evaluation target, its documentation and the evaluation results. Considering that meeting the accreditation requirements is the only prerequisite to provide evaluation and certification services at the "substantial" assurance level, a significant information security gap has been identified.

To cover the gap, ITSEF should seek independent confirmation that it is adequately managing information security. If market acceptance is essential for the ITSEF business model, such a management system should be based on the widely recognized international standard ISO/IEC 27001. The question is how to verify that all requirements are implemented correctly and perform as expected.

One option is to include the requirements of the Information Security Management System (ISMS) in the scope of accreditation. Unfortunately, this is not acceptable for the National Accreditation Body (NAB), as they cannot include in the scope of the accreditation audit requirements that could be subject to conformity assessment activities performed by entities covered by other accreditation programs. In this case, the ISMSs are certified by entities accredited to ISO/IEC 17021 and ISO/IEC 27006. NABs cannot accept the appearance of a conflict of interest. However, the NAB will respect a certificate confirming that the ITSEF ISMS is compliant with ISO/IEC 27001 if issued by an accredited certification entity.

The solution requires a huge workload for ITSEF to implement two management systems, one for laboratory activities and one for information security. However, taking into account the legal loopholes and formal constraints and following its original principles, NIT ITSEF has developed a unique approach to the management of its laboratory activities by defining one integrated management system that covers all topics, with a significant reduction of the efforts initially assessed. The concept is presented in [12]. The main steps to develop integrated management systems include the following:

- establishing an unified scope of both management systems,
- integrating security objectives with the primary process implemented in ITSEF,
- identifying standard components of the management system,
- identify parts of the management system that must be kept separately,
- implementing an appropriate system for documentation management,
- ensuring continuous support from top management,
- implementing awareness and training programs.

During the initial analysis, several parts are identified as the same or similar. These include:

- the context of the organization (ITSEF in this case),
- risks and opportunities in the management system,
- system procedures and related records (internal audits, management review, document control, corrective actions, continual improvement).

Then, several parts of the management system are closely related, and these include:

- information security and quality objectives,
- dealing with vendors and subcontractors,
- supporting assets, i.e. information systems, environmental and physical facilities,
- personnel competence management.

Finally, some parts of the management system should remain separate, and these include:

- ISO/IEC 27001: information security risk management, security of sites and IT facilities,
- ISO/IEC 17025: evaluation methodology and related activities.

The ITSEF integrated management system has successfully passed the relevant audits, and ITSEF is accredited (as of 2021) and certified (as of 2023). Experience gained during the maintenance of the integrated management system shows that the overhead for the integrated system is small. In 2024, when detailed requirements for the EUCC program were published [4], it became clear that the ISMS fully covers the extension of the accreditation requirements for information security implemented and successfully operating in the NIT ITSEF.

3.2. Challenges in Evaluations of Software Products with the Highest Attack Potential

The second important aspect of ITSEF readiness for the EUCC program is the ability to conduct evaluations of software products with the highest attack potential. Document [4] indicates two European standards that include protection profiles that require ITSEF to be used in evaluations with the highest attack potential (AVA_VAN.5).

The target of evaluation in the case under consideration was a software component of Trustworthy System Supporting Server Signing, which offers a remote qualified electronic signature as a service. The Signature Activation Module (SAM) component is responsible for authorizing the signing operation by checking whether: a) the signer authentication is correctly associated with the signing key and the data to be signed, and b) the signer is authenticated.

To ensure that the signer has exclusive control over the signing key, the signing operation is authorized by the SAM, which verifies a specific set of signature activation data (SAD) received from the signer via a dedicated application located on the server and activates the signing key in a cryptographic module (CM), both located in a protected environment. SAD verification means that the SAM checks the validity and integrity of the SAD elements and verifies that the signer is authenticated.

The SAM security specification and related security assurance requirements are included in the protection profile published in the European standard [13]. With respect to the security assurance requirements, the standard states that the vulnerability assessment should be performed with the attack potential indicated in the security assurance component AVA_VAN.5. Paper [13] does not provide any detailed methodology for developing appropriate penetration tests with such high attack potential.

Furthermore, no commonly recognized sources provide attack methods that could be relied upon for the evaluation of software products. Hence, the most challenging part of the ITSEF work was to develop tests and prove that the actual attack potential for these tests is equal to or higher than that indicated in the component AVA_VAN.5.

The starting point for the development of the methodology was the general approach to calculating the attack potential presented in [14]. Determining the attack potential corresponds to identifying the effort required to create an attack and demonstrating that it can be successfully applied to a specific object, thereby exploiting a vulnerability in that object.

When analyzing the attack potential required to exploit a vulnerability, the following factors should be considered:

- Time to identify and exploit (Elapsed time) – refers to the total time it takes an attacker to identify that a specific potential vulnerability may exist in targeted object, to develop an attack method, and to exert the effort required to attack that object.
- Technical expertise required (Specialist expertise) – refers to the level of general knowledge of the underlying principles, type of product, or attack methods (e.g., Internet protocols, Unix operating systems, buffer overflows).
- Knowledge of the design and operation of an object (Knowledge of the targeted object) – refers to specific specialized knowledge about the object (e.g., access to the source code and the ability of the evaluator to interpret and exploit it).
- Window of opportunity – refers to the identification or exploitation of a vulnerability that may require significant access to the target, which can increase the likelihood of detection. Some attack methods may require offline effort, and only short access to the target may be exploited. Access may also need to be continuous or over several sessions.

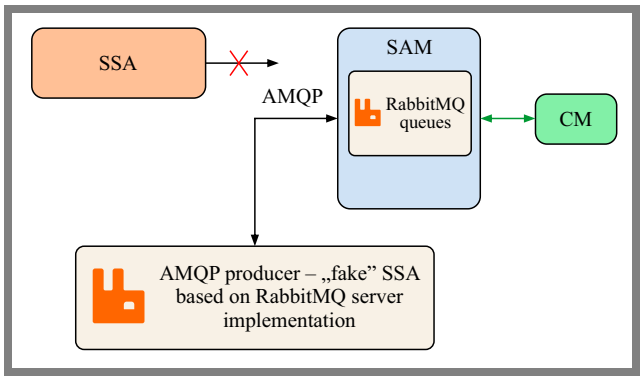


Fig. 1. Penetration test with the fake queue requester.

- IT hardware/software or other equipment required for exploitation – refers to the equipment required to identify or exploit the vulnerability (this may be standard, specialized, or bespoken equipment and generally measures equipment availability and cost).

Each factor is appropriately assessed, and an arithmetic value appropriate to the target of evaluation is assigned based on predefined rating tables. The attack potential is expressed as a score calculated by adding the values of all factors.

The general values given in [14] are intended to be replaced or refined according to the context (technology, type of product, etc.). Defining the set of values shared in each community is a non-trivial achievement. The leading CSPN framework [15] dedicated to pure software products has developed a set of factors and associated values derived from the general outline given in the specification [14].

The set of factors used in CSPN [15] is as follows:

- Time taken for the exploitation – it relates directly to the factor Elapsed time specified in [14],
- Attacker expertise – relates directly to the factor Specialist expertise specified in [14],
- Knowledge required by the attacker – relates directly to the factor Knowledge of the targeted object specified in [14],
- Access to the product by the attacker – it relates directly to the factor Window of opportunity specified in [14],
- Type of equipment needed – this factor is assumed from IT hardware/software or other equipment by simplifying the rating by using two levels: standard and specialized software tools.

In the performed evaluation, the methodology from [15] has been applied.

Furthermore, another reference source was considered to further validate the approach. The Common Vulnerability Scoring System (CVSS) methodology, described in [16], has been widely used by the IT security community for many years and is suitable for software products. When assessing the criticality of an identified vulnerability, one dominant factor called "attack complexity" is subject to rating. The "attack complexity" factor refers to the concept of the attack potential. It is defined as a metric that captures the measurable actions that must be taken by an attacker to actively avoid or bypass existing built-in security enhancement conditions in order to obtain a working exploit.

According to [16], when the attack complexity is considered "high", the successful attack depends on evasion or circumvention of security enhancing techniques in a place that would otherwise hinder the attack. The attacker needs to gather some knowledge about a specific target to carry out the final successful attack. To obtain specific information, the attacker must carry out additional attacks or otherwise break the security measures.

To present the methodology for calculating the attack potential, let us consider one attack developed for a given object. This attack is presented in detail in [17].

Tab. 1. Calculation of the attack potential for the test case.

Attack potential factor, based on the approach in [15]	Value	Score	Remarks
Time taken for (identification and) exploitation	$>$ 1 month	7	Two distinct types of software are to be investigated, and in-depth fuzzing is required
Attacker expertise	Multiple experts	8	The attack was needed to develop complex software
Knowledge required by the attacker	Critical	11	The source code was reviewed to find potential vulnerabilities
Access to the product by the attacker	Easy	1	Access to the front-end application as the user without any privileges
Type of equipment required	Specialized software	2	See the category "attacker expertise"
Total		29	Over 25, i.e. **Very High**

Tab. 2. The attack categories for the technical domain of "Hardware Devices with Security Boxes".

Attack category	Exemplary attack
Physical security invasive	Sensors removal and deactivation, removing and penetration potting materials, attack to an anti-tamper processor
Physical security semi-invasive	Perturbation test using a laser beam
Physical security non-invasive	Reverse engineering, power consumption analysis, emanation analysis, timing analysis
Electromagnetic and sound attacks	Monitoring keyboard sound or emanation, microwave scanning
Random number generation feature	Entropy analysis searching weaknesses
Software attacks off-device	Direct protocol attacks, man-in-the-middle and reply attack
Software attacks on the device	Secure operating system, hypervisor, virtual machine
PIN and cryptographic key-related	Limit key encryption key search by value, weakly padded PIN blocks

The penetration test aimed to deceive the authentication procedure provided by the SSA component and force the cryptographic module CM to sign an unauthenticated request from a fake signer. To do this, a fake queue requester was prepared and fake requests were made including false parameters (see Fig. 1). Another goal of the test was to force the service to crash and reject each request. The calculation of the attack potential relevant to the penetration test is presented in the Tab. 1.

It should be noted that the final calculation of the attack potential was further verified using the approach of [16] and the attack complexity value was evaluated at the level of "high". Therefore, the methodology for calculating the attack potential was shown to be correct and verifiable.

4. Future Work

The third work package is still under development. It aims to demonstrate the technical capabilities of NIT ITSEF in one of the two technical domains envisaged by [4] for the application for authorization.

The domain "Hardware Devices with Security Boxes" requires ITSEF be capable of performing the most advanced attacks with the attack potential of AVA_VAN.5. However, for such types of evaluated products, there is a set of state-of-the-art attack methods [18]. This means that ITSEF shall perform numerous attack categories, as shown in the Tab. 2.

The NIT ITSEF already covers most of the test methods presented in the Tab. 2 in the pilot evaluation required by [4]. Some of them still require additional effort to be completed and documented accordingly.

5. Conclusions

The requirements for ITSEFs that evaluate ICT products at the "high" assurance level [1] are challenging. AVA_VAN.5 is described in the Common Evaluation Methodology (CEM) [3] only in general terms, leaving room for certification schemes to specify detailed requirements that depend on technical domains or technologies.

The highest attack potential means that an appropriate methodology will be adopted, which on the one hand must be compliant with the CEM, but on the other hand must

be specific to the relevant attacks. The accreditation of test laboratories creates a significant advantage, since the basic principle of performing any tests under accreditation is the validation of the method and tool before testing.

The test laboratory management system, in the context of the EUCC, should include many security requirements due to the high sensitivity of the objects to be assessed and the test results. The best way is to integrate a management system for the quality and security of laboratory activities.

References

[1] European Parliament and the Council, *Regulation (EU) 2019/881 of the of April 17 2019*, No. 526/2013 (Cybersecurity Act) (https://eur-lex.europa.eu/eli/reg/2019/881/oj).

[2] Common Criteria for Information Technology Security Evaluation (CC:2022), Revision 1, November 2022 (https://www.commoncriteriaportal.org/index.cfm).

[3] Common Methodology for Information Technology Security Evaluation (CEM:2022). Revision 1. Standard developed by the Agreement on the Recognition of Common Criteria Certificates in the field of IT Security (CCRA). November 2022 (https://www.commoncriteriaportal.org/files/ccfiles/CEM2022R1.pdf).

[4] European Parliament and the Council *Commision Implementing Regulation (EU) 2024/482 of 31.1.2024* (http://data.europa.eu/eli/reg_impl/2024/482/oj).

[5] F. Bollman and K. Geyer, "Transition from National to the EUCC Scheme – BSI's Strategy for Supporting the Product Manufacturers and the ITSEFs during the Transition Phase", *2022 International Conference on the EU Cybersecurity Act*, Brussels, Belgium, 2022 (https://eucyberact.org/wp-content/uploads/2022/05/S22a-GeyerK.pdf).

[6] F. Bollman, K. Geyer, "Implementation of the EUCC Scheme in Germany: First Observations and the Way Forward", *International Conference on Cyber-Security & Resilience Act*, Brussels, Belgium, 2024.

[7] W. Slegers, "Implementation of and Transition to EUCC", *International Common Criteria Conference (ICCC'23)*, Washington DC, USA, 2023.

[8] Draft Accreditation of CBs for the EUCC Scheme, Version 1.6a, 2024 (https://certification.enisa.europa.eu/publications/draft-accreditation-cbs-eucc_en).

[9] Senior Officials Group – Information Systems Security, Mutual Recognition Arrangement (SOG-IS MRA), 2024 (https://www.sogis.eu/uk/mra_en.html).

[10] Common Criteria Recognition Arrangement (CCRA) (https://www.commoncriteriaportal.org/).

[11] J.M. Pulido, "2023 CC Certification Report", *International Common Criteria Conference (ICCC'23)*, Washington DC, USA, 2023.

[12] E. Andrukiewicz, "Unexpected Side Effect of the CSA – How CABs Could Demonstrate Their Competency in Information Security Area? ITSEF Use Case", *International Common Criteria Conference (ICCC'21)*, 2021.

[13] iTeh Standards, EN 419241-2:2019 – Trustworthy Systems Supporting Server Signing – Part 2: Protection Profile for QSCD for Server Signing.

[14] ISO, "Methodology for IT Security Evaluation", ISO/IEC 18045:2022 (https://www.iso.org/standard/72889.html).

[15] France's National Agency for the Security of Information Systems (ANSSI), "Procedure – Criteria for Evaluation in View of a First Level Security Certification", 2020.

[16] FIRST, "Common Vulnerability Scoring System version 4.0: Specification Document", 2024 (https://www.first.org/cvss/v4.0/specification-document).

[17] E. Andrukiewicz and P. Krawiec, "Use Case Related to the Software Product Evaluated with the Highest Attack Potential", *International Common Criteria Conference (ICCC'22)*, Toledo, Spain, 2022.

[18] Application of Attack Potential to Hardware Devices with Security Boxes Version 1.2, 2023 (https://certification.enisa.europa.eu/publications/application-attack-potential-hardware-devices-security-boxes_en).

Elżbieta Andrukiewicz, Ph.D.
ITSEF Manager
https://orcid.org/0000-0002-1030-6332
E-mail: e.andrukiewicz@il-pib.pl
National Institute of Telecommunications, Warsaw, Poland
https://www.gov.pl/web/instytut-lacznosci

Piotr Krawiec, Ph.D.
ITSEF Technical Manager
https://orcid.org/0000-0002-2395-5155
E-mail: p.krawiec@il-pib.pl
National Institute of Telecommunications, Warsaw, Poland
https://www.gov.pl/web/instytut-lacznosci

Techno-economics of IoT and OT Security

Morten Falch and Reza Tadayoni

Aalborg University, Copenhagen, Denmark

https://doi.org/10.26636/jtit.2025.FITCE2024.2032

Abstract — This paper provides an overview of the techno-economics of cybersecurity in IoT and OT devices. The purpose is to identify and provide justification for regulatory action within the area.

Keywords — cybercrime, cybersecurity, IoT, justification for regulation, OT, techno-economics

1. Introduction

Cybersecurity has become a serious challenge for businesses around the world. PricewaterhouseCoopers (PwC) has reported cybercrime to be the most widespread kind of economic fraud [1]. Cryptocurrencies valued at more than 400 million dollars were paid to ransomware addresses in 2020. This represents a growth of more than 400% in one year. At the same time attacks by malware increased by 358%. Distributed denial of service (DDOS), ransomware and other kinds of cyberattacks are happening more and more frequently, and for businesses this can lead to severe consequences, e.g. interruption of work processes and customer services, loss and compromising of data, violation of data protection and privacy laws, a lot of time wasted, and large additional costs.

The ongoing process of digital transformation is affecting all businesses and organizations, large and small, and this puts further focus on the challenges related to cybersecurity. In the latest global risk report published by the World Economic Forum the issue of cybersecurity reappeared to be among the top 10 global risks, and cyberattacks on critical infrastructure were seen as one of the risks with the largest potential impact on a global scale [2]. This concern is partly due to cyberattacks against Ukraine in 2022. Also cyberattacks jeopardizing privacy of vulnerable citizens is seen as a global risk.

In this regard Internet of Things (IoT) and Operational Technologies (OT) security are becoming still more important. The number IoT devices has exploded within the past decade and many of these are not sufficiently protected. A lot of IoT devices lack built-in capabilities for updating software and this makes it difficult to maintain security. Hackers cannot only hamper their functionality but can also use them as a gateway to other IT systems and devices. Especially badge readers, cameras and printers are devices of concern from a security perspective.

Likewise, OT security has gained in importance, as this is a key issue for securing critical infrastructures. Compared to the number of IoT devices, OT devices are lower in numbers, but more valuable. Many critical infrastructures are highly dependent on OT devices and disruption of their operations may have a detrimental impact on the functions of the society.

Cybersecurity as a policy issue has attracted a lot of attention both from a regulatory perspective and in economic literature. The EU has published a common strategy on cybersecurity [3] and several major initiatives are being launched by the EU to increase awareness and to protect critical infrastructures, e.g., the Network and Information Security 2 (NIS2) directive [4]. Likewise in the US the Executive Order 14028 is issued to protect critical infrastructures. Another legislation, which is relevant for cybersecurity of OT and IoT, is the forthcoming EU Cyber Resilience Act (CRA) [5]. This act will impose demands on cybersecurity for manufacturers of hardware.

The economics of cybersecurity is a relatively new area of research. While much research has been published on development of technical solutions and strategies for implementation, the economic foundation of any regulation or strategy for remedying cybercrime is still under development. This is especially the case when it comes to cybersecurity in IoT and OT devices.

This paper provides an overview of the IoT and OT security challenges, the techno-economic characteristics of possible cybersecurity measures to be taken, and the market failures to be addressed. Finally, the paper identifies the regulatory challenges that follows from this analysis. More specifically the paper discusses cybersecurity issues related to IoT and OT, how they can be addressed by the market, and where regulatory intervention is needed.

First the paper identifies IoT and OT cybersecurity challenges [6]–[8]. Then the economic characteristics of different security measures is discussed. As point of departure, this discussion is based on current research on cybersecurity as an economic good including [9]–[11]. Compared to these contributions, the authors take an approach, where the characteristics of the specific security measures identified in the technical analysis of IoT and OT are taken into account. Based on this, regulatory challenges regarding market intervention are identified.

2. Information Security, ICT Security, and Cybersecurity

Before we discuss the economic characteristics of cybersecurity, we will define the term cybersecurity and compare it with similar terms used in in the literature. Many of the contributions in cybersecurity economics are using a similar approach to what is applied in information security economics

This work is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) License.
For more information, see https://creativecommons.org/licenses/by/4.0/

and economics of privacy [11]. Here cybersecurity is seen as a collection of tools, which can be applied to protect information. It is therefore relevant to highlight how cybersecurity relates to these other terms. What are the similarities and differences and what are the implications of this on the economic characteristics?

Many definitions of cybersecurity are based on the so-called CIA triad: confidentiality, integrity, and availability of information. The CIA triad dates back to the 1970s, where it was introduced in [12]. The triad is also included in the definition provided by The International Telecommunications Union (ITU) [13]. This definition also specifies the kind of assets to be protected, namely connected computing devices, personal infrastructure, applications, services, and telecommunications systems. The CIA triad has later been complemented by non-repudiation, accountability, authenticity, and reliability of information. However, it is argued that this definition needs to be updated to include broader aspects of cybersecurity than just the technical protection of information [14].

Paper [13] makes a distinction among information security, ICT security, and cybersecurity based on the kinds of assets to be protected. Information security deals with the protection of information. Information might be stored or transmitted using ICT, but this is not necessarily the case. ICT security deals with the protection of the ICT system, which is used to store and handle the information. In contrast to this, cybersecurity is not always about confidentiality, access, or integrity of information. It also encompasses protection of non-information assets such as home automation systems and public utility infrastructures. Thus, cybersecurity can be defined as [13]:

"the protection of cyberspace itself, the electronic information, the ICTs that support cyberspace, and the users of cyberspace in their personal, societal and national capacity, including any of their interests, either tangible or intangible, that are vulnerable to attacks originating in cyberspace".

Article [14] argues that cybersecurity is more than just protection, and refers to the NIST framework, that includes five different activities: identify, protect, detect, respond, and recover. In this framework cybersecurity is more than just a product and includes an organizational framework to be implemented in order to protect the assets. This calls for a human-inclusive approach including sociological and psychological aspects, challenging the machine focused definition of cybersecurity [15]. Here the definition offered by [16] becomes relevant, as it offers a process-oriented view. Here cybersecurity is defined as "standard practices that involve the people, processes, and technologies in an organization, in a group, or stand-alone environments in which the computers and cyber-physical systems with valuable data are connected to cyberspace".

It follows that although there is a considerable overlap one must distinguish between the terms information security and cybersecurity. Information security deals with all kinds of information, also information not stored in a digital format. Cybersecurity deals only with digital information, but it includes as well also other kinds of assets such as computer systems and non-digital assets, which depend on the functioning of ICT based systems. Moreover, cybersecurity is not only about technical measures for protection. It is also about human and business processes [17].

Privacy is another term that is used in connection with cybersecurity. Privacy deals with personal information only and can be considered as a subset of information security. Moreover, privacy focuses on the confidentiality aspect and to a certain degree on the integrity aspect of the CIA triad. Still the economics of the three areas cybersecurity, information security, and privacy are closely related areas although they present distinct areas of research.

3. OT and IoT Security

OT and IoT technologies are facing a number of challenges when it comes to security and privacy issues. There are some specific risk factors and requirements related to these types of devices. It can be due to the limited computing and storage capacities and constrains on the power supply and battery capacity in the lightweight devices, when it comes to IoT, or in the way the equipment is integrated into the IT systems, when it comes to operational technologies (OT). The computing capacity and limited power supply make it difficult to develop advanced encryption protocols in the devices and the lack of integration in the IT systems makes it difficult to update OT devices and other IT equipment.

One important security and risk issue of IoT and OT is the standardization and regulation [18], [19]. This can relate to both protocols and the way the default set up of devices are configured. Many IoTs come with a minimum of security functions implemented and some come with default login and passwords without any requirement from the vendors that the user must change this default password before the use of the devices. This induces a big security risk. To avoid these security challenges, harmonized standards and regulatory initiatives at device and operator levels can be essential.

Another challenge is the legacy issues. The use of open access networks has exposed the supervisory control and data acquisition (SCADA) systems to cyber attacks [20]. Many IoT and OT devices are based on old software and hardware frameworks, which are difficult to update to adopt to modern security and privacy standards and requirements without allocation of enormous financial resources. For example, when it comes to the OT, the SCADA systems, which are used in many critical infrastructures, are old and imply significant security risks. This issue was raised already back in 2006 "the increasing interconnectivity of SCADA networks has exposed them to a wide range of network security problems" [21]. One of the major vulnerabilities of the OT systems based on SCADA is the growth of connectivity of internal company networks to the outside word resulting in possibilities for cyberattacks etc. Many OT devices are not updated properly as this will demand down time in the line of production, and in some cases the device and equipment are not integrated in the IT environment of the company, which makes it difficult to be updated.

The physical accessibility is another challenge when it comes to the IoT and OT devices, as the devices may be easily accessible from outside and often without proper surveillance [22]. Of course, proper security strategies and allocation of the necessary financial resources to prevent physical accessibility can be restricted to critical devices, but legacy systems have not prioritized this aspect and continue to be a challenge.

Another important security risk, which is a hot topic for the time being, is supply chain attacks, which can be made by comprising the software or hardware in a specific vulnerable part of the value chain. Vulnerabilities can come from the physical access to part of the value chain or weak access mechanisms when uploading software or introducing new hardware. SolarWinds hack is a prime example of a supply chain attack [23], [24].

Furthermore, the regular cyber security risks like ransomware attacks, where critical IoT or OT devices are locked by criminals [25], and DDoS attacks, where huge amounts of IoT devices are used to send great quantities of requests and thereby overload the receiving systems and put them out of function [26], are examples of exploitation of the vulnerabilities of IoT and OT systems.

A last thing we want to mention is the privacy issues [27]. The IoT devices gather huge amount of data. Some of these may be sensitive company data or include personal information of users or customers.

Solutions to all the abovementioned problems include an interplay of technological, economic and regulatory aspects. We need new technologies and security strategies at device and infrastructure levels. But we also need financial resources and new business models as well as policy and regulatory interventions to create mandatory security standards and practices.

4. Research in Economics of Cybersecurity

Article [16] provides an extensive literature review of different research directions on cybersecurity economics. The review is based on 28 studies selected among more than 600 models identified by the authors:

1) Budgeting finding the optimal level of investments in cybersecurity
   - Investment,
   - Externalities,
   - Insurance.
2) Economic efficiency
   - Misallocation of resources,
   - The type of good (private, common, club, public).
3) Interdependent risks
   - Network effects,
   - Lock-in effects,
   - Supply-chain risks.
4) Information asymmetry
5) Governance
6) Cybercrime
7) Sustainability

There is a considerable overlap between most of these topics. While the first point deals with cybersecurity at the micro-level, where the possible action of the individual organization is the point of departure, the remaining points address issues at the meso and macro levels.

Although the list includes diverse research issues, the overall theme is how to decide the optimum level of investments in cybersecurity and the scope for public intervention. These issues are related to the economic characteristics of cybersecurity and the lack of transparency of the market.

The issue of cybercrime looks in principle at the same issues, but here the focus is on the market, where the cybercriminals act. How is the market for cybercrime structured and what are the economic characteristics of the products offered on this market?

The economic characteristics are especially related to research on externalities of cybersecurity products. Investments in cybersecurity may imply strong positive externalities, as they may prevent spreading of malware etc. beyond the stakeholder financing the investment.

Borrowing from information economics, Samuelson's concept of public good is often used for describing the economic characteristics of cybersecurity. Samuelson distinguishes among four types of goods according to two parameters (rivalry and exclusivity (Fig. 1): normal goods, club goods, common goods, and public goods. Tangible goods such as foodstuff, cars, computers etc. are rivalrous as they can be consumed only once. They are also excludable as access is limited. These goods are termed normal goods.

Information on the other hand can be used many times. Therefore, information goods are seen as being non-rivalrous. Depending on the context, information goods are in principle also non-exclusive, as they can easily be copied and made available to everybody, as soon as the information is revealed. Information is therefore termed as a public good, along with a range of public services offered by the government. National defense is the most prominent example of a public good. It is however possible to restrict access to certain information goods. In this case they can be termed as club goods.

When it comes to cybersecurity, several authors see this as a product with strong public good characteristics [16], [28].

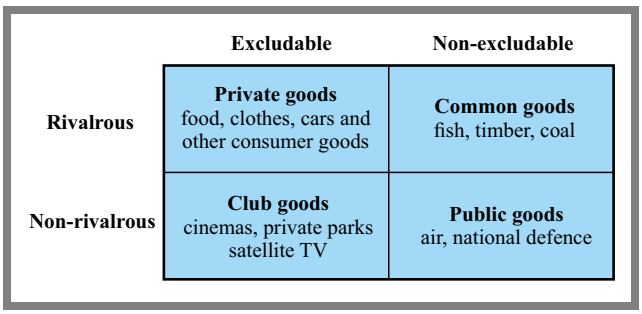


Fig. 1. Private goods, common good, club good, and public goods.

The argument is primarily the strong positive external effects that investments in cybersecurity may have on other actors. Another reason may be that cybersecurity can be considered as a kind of information good with similar economic characteristics as other information goods. Therefore, the positive externalities are often taken for granted. When consumption of public goods is up to an individual decision-making on an unregulated market, this will result in underinvestment.

However, it can be problematic to treat cybersecurity as one single homogeneous product. Achieving cybersecurity takes investments in a wide range of different measures, each with their own economic characteristics.

One approach to go a bit deeper into the economic characteristics of cybersecurity is to analyze cybersecurity for different types of actors and how they interact. Bauer and Eeten claim that externalities include spill-over effects among different types of actors and provide a framework for analysis of these spill-over effects [28]. Their analysis focuses on analysis of cybersecurity products implemented at the network level, and their impact on security for other groups of actors. Following groups of actors are included in the analysis: ISPs, application and service providers (App/Svc), hardware and software vendors, users, security providers, and national and international organizations.

The ISPs constitute the core of the ecosystem. ISPs are interconnected and their level of cybersecurity is highly dependent on the security level in those ISPs to which they are connected. Moreover, they depend on application and service providers, security providers, hardware and software vendors, and users. Finally various governance institutions may contribute to the level of security. The point made in this paper is that each actor will decide on the level of investments according to their own costs and benefits, and free riders may occur. Some spill-over effects may be reflected in the prices. For instance, may users be willing to pay for having an ISP they consider offering a high level of cybersecurity. However, the market for cybersecurity is far from being transparent and information asymmetries exist.

With regard to IoT and OT, it is important also to look at cybersecurity achieved at the device level. Here equipment manufactures and standardization bodies are important actors.

Supply chain risk is another kind of spill-over effect. Here companies are attacked via their suppliers. These may include small companies with little protection. Attacks may be made via connections to IoT or OT devices with insufficient protection owned by these companies.

Economic models estimating costs and benefits are made with the purpose of finding the optimum investment level for cybersecurity. Most of these models use security level as an aggregated economic variable [16]). Thus, the models provide little guidance in the kinds of security products, which are the most attractive to invest in.

The research topic cybercrime includes mainly estimation of costs incurred in companies attacked and economic consequences at meso- or macrolevels. This relate to the budgeting, as it relates to estimation of benefits to be achieved by investing in cybersecurity. The economics of the cybercriminals and their business models seem to be a different topic, which is excluded from the framework provided by [16]. The economics of cybercriminals is however important for a study on the economics of cybersecurity, as the key aim of cybersecurity products is to make current business models for cybercrime unviable and prevent creation of new viable business models. Research in this topic, which is truly interdisciplinary as technical and economic analysis needs to be combined, seems to be published primarily in engineering fora.

5. Categorization of Cybersecurity Products

Cybersecurity products include a wide range of activities carried out with the purpose of protecting an organization against cybercrime. The National Institute of Standards and Technology at the U.S. Department of Commerce (NIST) has developed a framework for what to be done in order to be protected [29]. This framework is also used in Europe, where ISO has developed international standards (ISO 27001 and ISO 27002) based on the same principles. The framework includes five core functions, which should be addressed:

- Identify includes identification of the critical processes and resources. This includes all kinds of IT and IoT devices, software, and data. Especially sensitive data, for instance personal information, and data critical to the operations of the company should be included. Moreover, roles and responsibilities for employees, vendors, and others with access to sensitive data must be identified.
- Protect includes protection of the facilities and sensitive data identified above. Some protection is built into the standard software applied by companies. Still many security measures in particular organization and human measures are up to the individual organization to implement. Email filters with blacklisting or even whitelisting can help to avoid phishing and emails with harmful content to be opened, but awareness of employees is even more important in this respect. Moreover, access to any system should be restricted as much as possible.
- Detect includes detection of cybersecurity attacks. IT systems must be monitored in order to detect any cybersecurity events as early as possible. This includes unauthorized access and unusual traffic patterns.
- Respond includes guidelines for how to react if a cybersecurity attack is detected. and how to limit damages. An early response from the user of an infected machine may prevent potential damages to be spread to other parts of the IT system itself, as well as damages in of facilities hosted by trading partners or elsewhere. Trading partners and authorities should be informed about cybersecurity event.
- Recover includes guidelines for reestablishment of damages made in an attack, and reestablishment of data, systems, and business processes.

Cybersecurity as a product possesses some obvious positive external effects, as it is stated in most papers dealing with economics of cybersecurity. However, when looking at the different kinds of cybersecurity measures an organization can invest in, it follows that only a few of them possess notable externalities or spill-over effects, which relate to protection of other actors. These effects are in particular related to protection of facilities and detection of attacks, and concerns attacks of other organizations for instance through dissemination of malware.

In addition to these effects, substantial externalities may be related to possible interruptions in operations. This is a key issue for public utilities, for other public services, and even some private companies. Interruption in service delivery may be caused by many different kinds of incidents of which cybersecurity is only one.

6. The Market for Cybercrime

Looking at the market for cybercrime, it is important to distinguish between different types of hackers and their motives. Hackers are not always criminals looking for profit. Hackers can also be motivated by curiosity, recognition or revenge. Many papers on cybersecurity provides definition of the types of hackers and their motivations. [30] provides an extensive overview of the different definitions and suggests a categorization with 15 different types of hackers. In this context, the key issue is whether a hacker attacks a specific company or if they attack any company, which is vulnerable for a cyberattack. Moreover, it is also important, whether the attack harms other parties. If some strategic information is stolen from a specific company, it will probably only harm the specific company and the externalities are limited. However, if financial information on banking customers is stolen, e.g. from a financial institution, this will affect many different actors outside the company.

Hackers are using a wide range of methods to attack companies, and the economic characteristics of cybersecurity depend on the kinds of attacks.

The European Union Agency for Cybersecurity, ENISA has in a report identified the following prime threats [31]:

- ransomware,
- malware,
- crypto jacking,
- e-mail related threats,
- threats against data leaks,
- threats against availability and integrity,
- disinformation – misinformation,
- non-malicious threats.

Ransomware is reported to be the most important thread, here attackers encrypt an organization's data and demand payment to restore access. If ransom money a paid, this may encourage similar attacks on other companies. Malware "intended to perform an unauthorized process that will have an adverse impact on the confidentiality, integrity, or availability of a system" [31]. Malware is also considered to be a prime threat. Malware can be spread from one company to another and this is the primary policy argument for implementing regulatory measures in order to ensure cybersecurity.

Crypto jacking where criminals steal computing power to generate cryptocurrency can hit any owner of a computer. An increase in this type of cybersecurity breach has been observed, but it will only harm the computer infected.

E-mail related threats are reported to be increasing in spite of educational campaigns to increase awareness. Infected e-mails and phishing e-mails can be sent to anybody, but in organizations with less formal procedures for data-handling and updating of filters are the most vulnerable. ISPs and other service providers can protect their customers through installation of various filters. The threat of leaks of sensitive data depends on the kind of data. Leaks of data belonging to companies or private persons imply spill-over effects on other actors, and this is one of the arguments for having rules on protection of personal data.

Availability and integrity of data can be compromised in different ways, of which denial of service and web-based attacks are the most important. According to ENISA, this threat ranks high. Here, the spill-over effects are important, as this kind of attacks involve the use of a botnet using infected devices connected to the Internet such as IoTs. The availability of unprotected devices is therefore a threat also for other actors. Disinformation and misinformation delivered through social media is on the rise. Non-malicious threats include threats, where the malicious intent is not apparent. These do not originate from cyber criminals or other types of hackers but are mostly based on human errors or misconfigurations. These issues go beyond the scope of this paper.

7. Conclusion

A decomposition of cybersecurity in IoT and OT devices into its different components reveals the kinds of externalities and spill-over effects that relate to cybersecurity. In this way the analysis can contribute to identification of regulatory needs and design of the right regulatory measures to be implemented.

Some externalities are caused by the specificities of the concept of cybersecurity, while others are more generic in nature. The latter ones relate to two different kinds of impacts:

- Payment of ransom money may encourage cybercriminals to continue their activities and help funding of investments in developing new tools for cyberattacks.
- Cyberattacks may lead to discontinuation of operations of the company subject to attack. If the target has been a critical infrastructure, this may have severe consequences also for other actors.

These two externalities are not related to a specific technology or a method applied by cybercriminals only to the outcome of the attack. Other kinds of impacts are more specific and depends on the kinds of attack:

- An organization may be possession of information, which is sensitive to other actors. Most important in this regard is personal information of private customers. In this case the costs of intrusion by cybercriminals may not be borne by the organization itself, but by their customers. This externality relates to information security, which overlaps with cybersecurity, and is addressed by privacy regulation, e.g. GDPR.
- Malware can be spread from one organization to another, if not properly protected. Therefore, there is a common interest in having a minimum level of protection in all devices connected to the Internet. This includes IoTs and networks operated by SMEs or private citizens.
- A special version of this is DDoS, where cybercriminals utilize their control of a large number of infected devices to create overload on specific systems. As discussed in this paper IoT and OT devices are often used for this type of attacks.
- Another variation is supply-chain attacks, where a business partner with a vulnerable network is used as a gateway for infecting well-protected systems.
- ISPs play a special role in this context, as they can offer improved protection to their customers. Thus, there are spill-over effects from one type of actors to another. This may be an argument for regulation if the market cannot provide the right incentives, for instance by having cybersecurity defined as a parameter for competition.

Finally, it should be noted that cybersecurity includes organizational as well as human factors in addition to technology. For instance, creation of awareness is a key tool when fighting against fishing. In this case information campaigns may be more efficient than regulation.

References

[1] PricewaterhouseCoopers, "PwC's Global Economic Crime and Fraud Survey 2022", *PricewaterhouseCoopers*, 2022 (https://www.pwc.com/gx/en/services/forensics/economic-crime-survey/2022.html).

[2] World Economic Forum, "Global Risk Report 2024", World Economic Forum, 2024 (https://www.weforum.org/publications/global-risks-report-2024/in-full/).

[3] European Commission, "Joint Communication to the European Parliament and The Council: The EU's Cybersecurity Strategy for the Digital Decade, JOIN (2020) 18 final", Brussels, 2020.

[4] European Commission, "NIS2 Directive", Brussels, 2020.

[5] European Commission, "EU Cyber Resilience Act, Shaping Europe's Digital Future", Brussels, 2022.

[6] M.M. Noor and W.H. Hassan, "Current Research on Internet of Things (IoT) Security: A Survey", *Computer Networks*, vol. 148, pp. 283–294, 2019 (https://doi.org/10.1016/j.comnet.2018.11.025).

[7] R. Mahmoud, T. Yousuf, F. Aloul, and I. Zualkernan, "Internet of Things (IoT) Security: Current Status, Challenges and Prospective Measures", *10th International Conference for Internet Technology and Secured Transactions (ICIT)*, London, UK, 2015 (https://doi.org/10.1109/ICITST.2015.7412116).

[8] W.A. Conklin, "IT vs. OT Security: A Time to Consider a Change in CIA to Include Resilience", *49th Hawaii International Conference on System Sciences (HICSS)*, Koloa, USA, 2016 (https://doi.org/10.1109/HICSS.2016.331).

[9] I. Brown, "The Economics of Privacy, Data Protection and Surveillance", in: *Handbook on the Economics of the Internet*, Edward Elgar Publishing, pp. 247–261, 2016 (https://doi.org/10.4337/9780857939852.00020).

[10] H. Asghari, M. van Eeten, and J.M. Bauer, "Economics of Cybersecurity", in: *Handbook on the Economics of the Internet*, Edward Elgar Publishing, pp. 262–287, 2016 (https://doi.org/10.4337/9780857939852.00021).

[11] A. Odlyzko, "Cybersecurity is Not Very Important", *Ubiquity*, pp. 1–23, 2019 (https://doi.org/10.1145/3333611).

[12] J.P. Anderson, "Computer Security Technology Planning Study", Technical Report for USAF, 1972.

[13] R. von Solms and J. van Niekerk, "From Information Security to Cyber Security", *Computer and Security*, vol. 38, pp. 97–102, 2013 (https://doi.org/10.1016/j.cose.2013.04.004).

[14] J. van der Ham, "Toward a Better Understanding of 'Cybersecurity'", *Digital Threats: Research and Practice*, vol. 2, pp. 1–3, 2021 (https://doi.org/10.1145/3442445).

[15] M.G. Cains *et al.*, "Defining Cyber Security and Cyber Security Risk within a Multidisciplinary Context Using Expert Elicitation", *Risk Analysis*, vol. 42, pp. 1643–1669, 2022 (https://doi.org/10.1111/risa.13687).

[16] M. Kianpour, S. Kowalski, and H. Øverby, "Systematically Understanding Cybersecurity Economics: A Survey", *Sustainability*, vol. 13, art. no. 13677, 2021 (https://doi.org/10.3390/su132413677).

[17] A. Sarri, V. Paggio, and G. Bafoutsou, "Cybersecurity for SMEs - Challenges and Recommendations", European Union Agency for Cybersecurity (ENISA), Heraklion, Greece, 2021.

[18] P. Radanliev *et al.*, "Future Developments in Standardisation of Cyber Risk in the Internet of Things (IoT)", *SN Applied Sciences*, vol. 2, art. no. 169, 2020 (https://doi.org/10.1007/s42452-019-1931-0).

[19] I. Brass *et al.*, "Standardising a Moving Target: The Development and Evolution of IoT Security Standards", *Living in the Internet of Things: Cybersecurity of the IoT*, London, UK, 2018 (https://doi.org/10.1049/cp.2018.0024).

[20] S. Ghosh and S. Sampalli, "A survey of security in SCADA networks: Current issues and future challenges", *IEEE Access*, vol. 7, pp. 135812–135831, 2019 (https://doi.org/10.1109/ACCESS.2019.2926441).

[21] V.M. Igure, S.A. Laughter, and R.D. Williams "Security Issues in SCADA Networks", *Computers and Security*, vol. 25, pp. 498–506, 2006 (https://doi.org/10.1016/j.cose.2006.03.001).

[22] E. Schiller *et al.*, "Landscape of IoT Security", *Computer Science Review*, vol. 44, art. no. 100467, 2022 (https://doi.org/10.1016/j.cosrev.2022.100467).

[23] R. Alkhadra, J. Abuzaid, M. AlShammari, and N. Mohammad, "Solar Winds Hack: In-depth Analysis and Countermeasures", *12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Kharagpur, India, 2021 (https://doi.org/10.1109/ICCCNT51525.2021.9579611).

[24] M. Willett, "Lessons of the SolarWinds Hack", *Survival. Global Politics and Strategy*, vol. 63, pp. 7–26, 2021 (https://doi.org/10.1080/00396338.2021.1906001).

[25] M. Al-Hawawreh, E. Sitnikova, and N. Aboutorab, "Asynchronous Peer-to-peer Federated Capability-based Targeted Ransomware Detection Model for Industrial IoT", *IEEE Access*, vol. 9, pp. 148738–148755, 2021 (https://doi.org/10.1109/ACCESS.2021.3124634).

[26] R. Vishwakarma and A.K. Jain, "A Survey of DDoS Attacking Techniques and Defence Mechanisms in the IoT Network", *Telecommunication Systems*, vol. 73, pp. 3–25, 2020 (https://doi.org/10.1007/s11235-019-00599-z).

[27] L. Tawalbeh, F. Muheidat, M. Tawalbeh, and M. Quwaider, "IoT Privacy and Security: Challenges and Solutions", *Applied Sciences*,

vol. 10, art. no. 4102, 2020 (https://doi.org/10.3390/app10124102).

[28] J.M. Bauer and M.J. van Eeten, "Cybersecurity: Stakeholder Incentives, Externalities, and Policy Options", *Telecommunications Policy*, vol. 33, pp. 706–719, 2009 (https://doi.org/10.1016/j.telpol.2009.09.001).

[29] M.P. Barrett, "Framework for Improving Critical Infrastructure Cybersecurity, Version 1.1", NIST, 2018.

[30] S. Chng, H.Y. Lu, A. Kumar, and D. Yau, "Hacker Types, Motivations and Strategies: A Comprehensive Framework", *Computers in Human Behavior Reports*, vol. 5, art. no. 100167, 2022 (https://doi.org/10.1016/j.chbr.2022.100167).

[31] ENISA, "ENISA Threat Landscape 2021", 2021 (https://www.enisa.europa.eu/publications/enisa-threat-landscape-2021).

Morten Falch, Ph.D., Assoc. Professor
Department of Electric Engineering
https://orcid.org/0000-0002-2649-215X
E-mail: falch@es.aau.dk
Aalborg University, Copenhagen, Denmark
https://www.en.aau.dk

Reza Tadayoni, Ph.D., Assoc. Professor
Department of Electric Engineering
https://orcid.org/0000-0003-2217-0919
E-mail: reza@es.aau.dk
Aalborg University, Copenhagen, Denmark
https://www.en.aau.dk

The Potential Cyber and Network Security Issues of PSTN Closure

Andy Valdar

University College London, London, United Kingdom

https://doi.org/10.26636/jtit.2025.FITCE2024.2021

Abstract — Up until a few years ago, all phone calls over land lines, mobile networks, cable TV networks and many altnets used circuit-switching technology. This has been the case despite the massive build-up of packet-based data networks – and the dominance of Wi-Fi, broadband, and the Internet in people's lives over the last 20 years or so. Now all these network operators are engaged in shifting telephone service onto their packet-based data infrastructures and withdrawing the obsolescent circuit switched technology. This article considers why this change is happening, how calls will be handled in the future and the big challenges faced by landline operators in this transition, with special emphasize on the potential cyber and network security issues involved.

Keywords — all-IP NGN, IP phone, PTSN, VoIP

1. The End of an Era

Since its introduction in the late 1800s the switching of land-line telephone calls within the public switched telephone networks (PSTN) has relied on a succession of systems based on the best technologies of the day. Figure 1 shows a stylized view of the types of technology that have come and gone over the last century Given the often-lengthy transition periods as old systems are replaced by the new, invariably at any one time there will be a mix of different technologies in a PSTN. Despite this, network operators have provided continuity of service, upgrading exchange systems with little or no breaks over the years. Up until recently all telephone switching systems – originally manual then automatic analogue and now digital time-division multiplexed (TDM) electronic type equipment – have been circuit switched, providing "connection-orientated" continuous bi-directional paths between calling and called subscribers for the duration of the call [1].

Now, PSTNs around the world are currently moving into the new era of "connectionless" digital packet switching for telephony. Unlike the existing digital TDM systems which are synchronous, digital packet systems are asynchronous. In this context, synchronous means that the encoded voice bits – conveying speech and silence – are continuously transmitted and switched in both directions within the circuit under the control of regular clock pulses. Asynchronous means that encoded voice bits are grouped into appropriate packets and forwarded through the network routers on an as-and-when basis. The packet approach has the important advantage that the voice packets can be easily interlaced with data packets

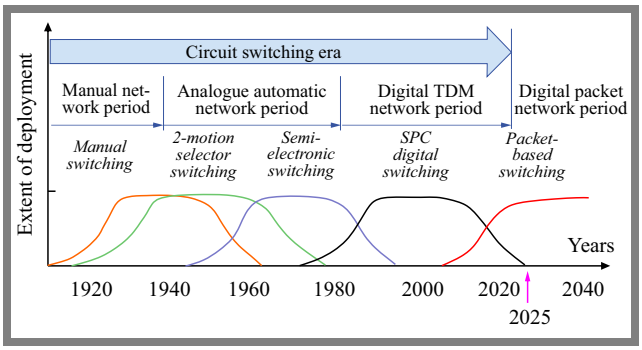


Fig. 1. Telephone switching technology life cycles.

on the same highway, so creating an integrated voice-data network (or multi-service platform). However, the asynchrony also means that packets can experience variable delay across the networks which can lead to impairment of the speech quality. The likelihood of such problems being perceived by the listeners is minimized by appropriate dimensioning of network capacity and giving higher priority to voice packets.

2. Why Change?

Stored-program-controlled (SPC) digital switching systems were introduced into PSTNs around the World during the 1980s. Notable examples include the US ESS, the Japanese NEAX, the German EWSD, the French System-12, the Canadian DMS100, the Swedish AXE10, and the British System-X. By the turn of the century, most of these systems were reaching the end of their economic life. Although generally still working well, manufacturers progressively began to reduce their support, so availability of spare parts and software upgrades became a problem for network operators. It was clear that replacements would be needed eventually, but the big question was what technology should be used to replace these exchanges?

The telecommunication environment had changed considerably since the SPC digital circuit switches were first introduced given the huge rise in digital data traffic generated by both businesses and consumers. Known in the industry as the "data wave", it was estimated that in the UK, the number of bits carrying data exceeded that of digital pulse-code modulation (PCM) encoded voice around 2005 for the fixed network, and 2010 for mobile networks. Equally important, packet-based network equipment was expanding beyond enterprise

This work is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) License.
For more information, see https://creativecommons.org/licenses/by/4.0/

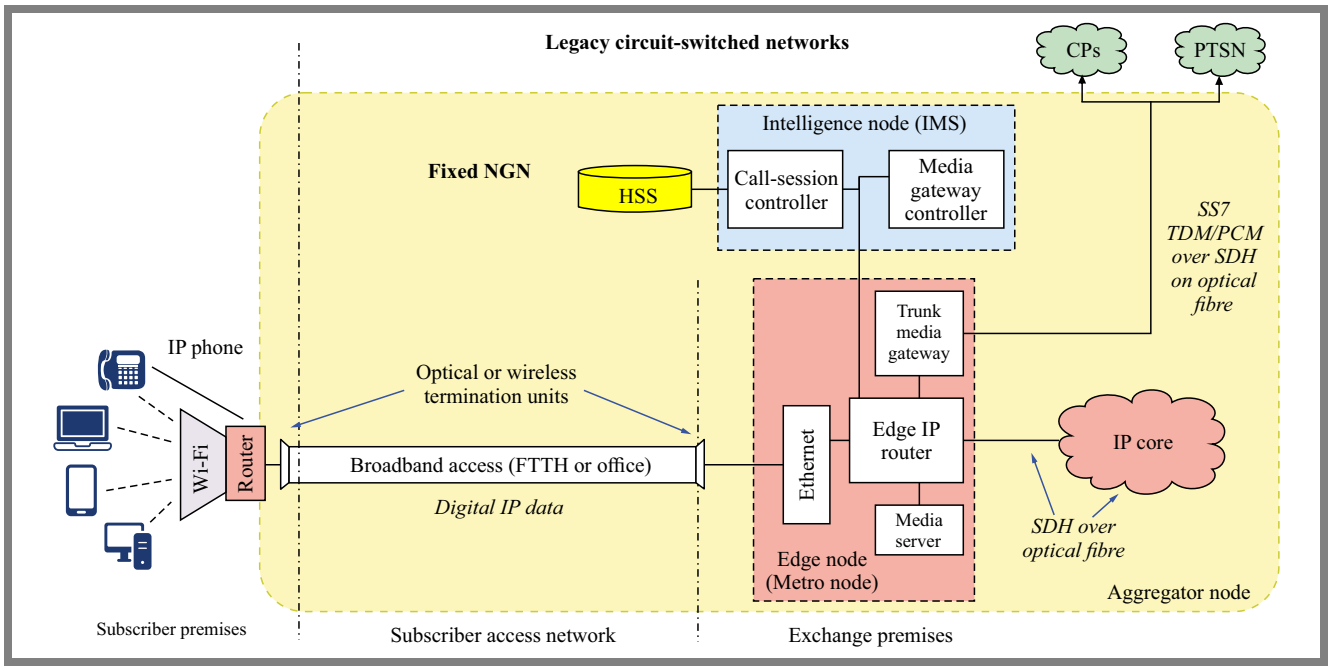


Fig. 2. Generalized view of an NGN replacement of PSTN exchanges.

networks and becoming a credible alternative to the existing equipment used in the PSTN and public data networks (i.e., carrier grade). This led to the International Telecommunications Union (ITU) in 2006 defining the concept of the Next generation network (NGN), which characterized the features of future networks that would not only replace the PSTNs but also form a common platform for an operator's mobile and data networks. [2] They also identified the possible scenarios for transforming the existing PSTN to an NGN, which are considered later.

The key characteristics of an NGN are an Internet protocol (IP) platform which supports voice and data services with both fixed and mobile access to the customers. This means that the telephony service currently on the PSTN will be handled as voice-over IP (VoIP) instead of being circuit-switched, and the signaling throughout the network would be session initiation protocol (SIP) instead of signaling system No. 7 (SS7). An important difference between the VoIP and data services provided by an operator's NGN and similar services currently carried over the Internet (e.g., over-the-top applications such as WhatsApp) is that the NGN is a managed platform in which appropriate quality for the various voice and non-voice services can be maintained. As well as providing a viable replacement for the PSTN, the industry expected that the use of multi-service IP platforms would enable new multimedia services, combining digital data, voice, and vison. The expectation was that by moving services onto a common IP platform the operator's many service-specific networks can be closed, giving operational cost savings due to having to operate just one network.

The internationally agreed ITU recommendations on NGNs gave the industry – operators and manufacturers – a defined target to replace the PSTNs. However, it has taken many years for the IP-based equipment to become sufficiently carrier grade to cope with the scale and complexities of the PSTN, with its wide range of users' access lines, services other than telephony, business, and residential users' terminals, etc. In addition, local circumstances, different levels of telecom market competition, national regulation and government policies have influenced the rate of adoption of the NGN concept in each country. Finally, making a credible business case for the huge investment required to replace the PSTN has proved be a big challenge. However, from around 2010 onwards operators have embarked on making the transition to an NGN.

Of course, in many countries there are several operators providing telephone service: the incumbent operator and a few alternate operators, including some cable TV companies. Many of the alternate operators have their own circuit-switched PSTN. Interestingly, around the world the programs of withdrawing the PSTNs have been publicized using a variety of names – "PSTN sunset" or "POTS switch-off" in the USA, "End of the PSTN" in France, part of the "National Broadband Network" rollout in Australia, "All-IP transformation" in Germany, and "The move to all-IP" in the UK.

3. What does the Replacement for the PSTN Exchange Look like?

The NGN replacement for the PSTN has a completely different architecture. There is no one-for-one replacement of the circuit digital TDM switch-blocks. Instead, packetized voice is carried through a data Core Network of IP routers through to the destination subscriber's line under the control of multiple intelligence nodes, usually dispersed across the network. The call is set up through the router network using a sequence of

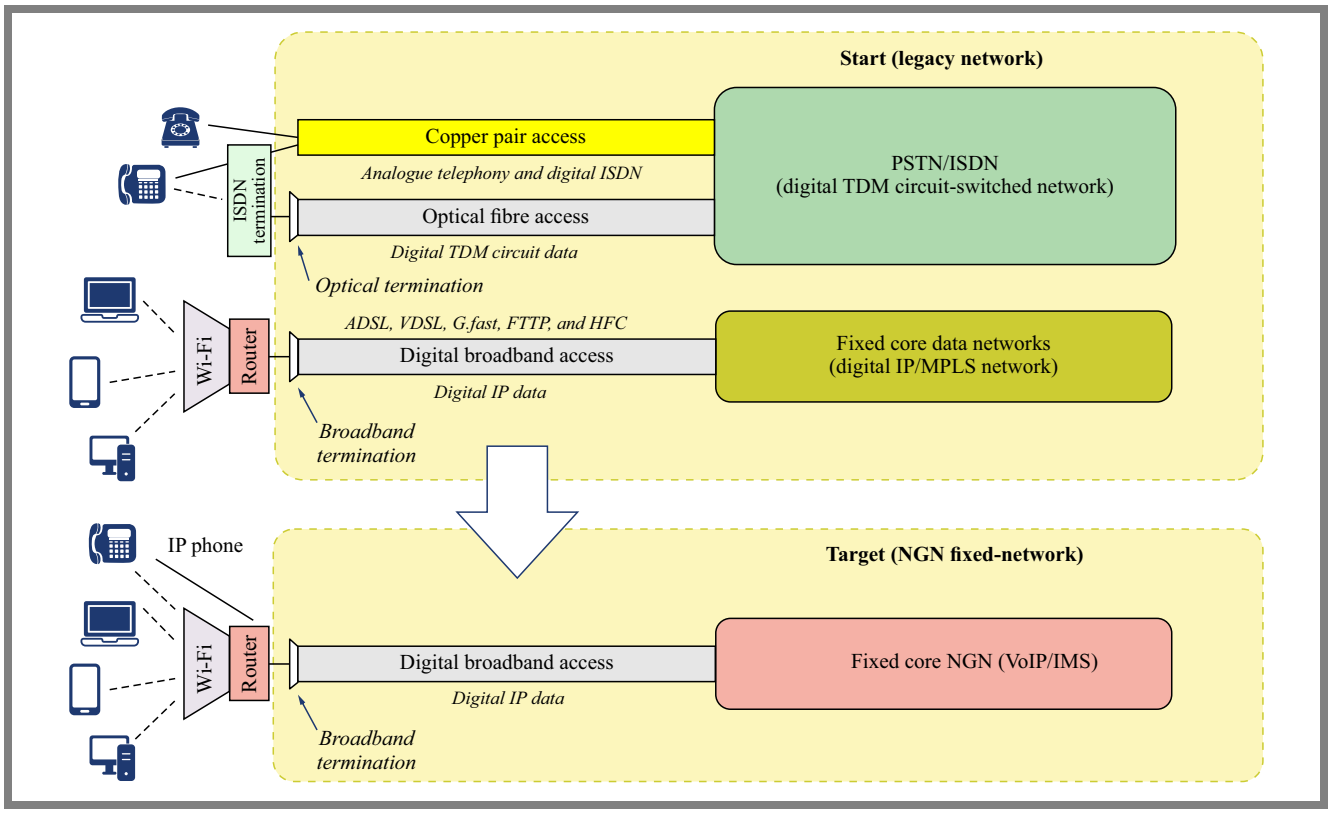


Fig. 3. Generalized view of the transition to all-IP NGN.

SIP messages. Ringing tone is inserted into the audio channel at the called subscriber node, to give users reassurance that the call has succeeded.

A simple schematic diagram of an NGN network is shown in Fig. 2. It is assumed that the subscribers' telephones, whether business or residential, are IP (VoIP) phones, with voice codec, conversion to/from IP packets and SIP signaling within the instrument. The IP phone is connected to the access broadband line system either directly or via Wi-Fi. At the exchange building the broadband line system is linked to an Ethernet switch or an edge router, which provides the classic edge network function of traffic aggregation.

Voice packets are routed through the operator's IP network interlaced with the various non-voice data packets. The control of the telephony service is provided by the call-session control function (CSCF) in an intelligence node, usually remote within the network. Subscriber's profile data, such as telephone numbers, service features, etc., are stored in the home subscriber server (HSS) and there is also a call charging system which supports the "calls" and stores the call records (for simplicity not shown in Fig. 2).

Interestingly, routing is still based on telephone numbers. Therefore, the dialed number first needs to be converted from the recipient's phone number to an appropriate IP address (actually, a universal resource identifier, URI) for insertion into the headers of the sender's voice packet. This conversion generally uses the ENUM algorithm [3] in a domain name system (DNS) within the network (not shown in Fig. 2).

Finally, the trunk media gateway provides an interworking facility between the NGN and remaining circuit-switched (usually known as legacy) networks to which it needs to connect, nationally or internationally. This gateway provides both voice transcoding between IP packets and TDM PCM channels, as well as signaling conversion between SIP and SS7 messages. Where necessary, the interworking may also involve embedding certain service-specific SS7 messages within the SIP IP packets (i.e., embedded ISUP).

It is now generally accepted that the control functions in the intelligent node should adhere to the internationally standardized IP multi-media subsystem (IMS) architecture [4]. This system is specified to facilitate a wide range of IP-based services on fixed and mobile networks. For example, IMS has recently been deployed to enable voice switching on 4G mobile networks – i.e., voice on LTE (VoLTE). Although capable of supporting many types of multi-media (voice, data, and video) services, the biggest current application is still telephony. By making IMS part of the NGN replacement for the (fixed) PSTN, operators can progressively build a unified NGN platform for fixed and mobile access covering voice and data services. This produces platform integration as well as service convergence, as envisaged by the ITU recommendations.

4. The Transition to an All-IP Network

The big challenge facing network operators worldwide has been deciding how to replace large numbers of PSTN ex-

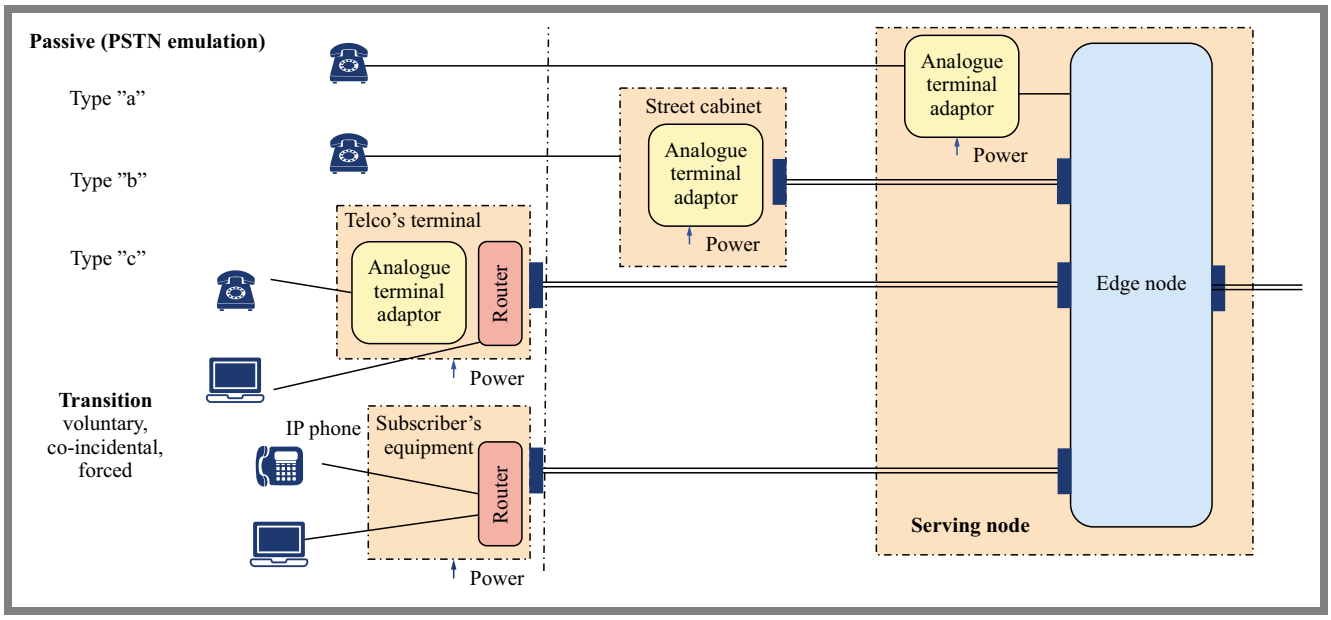


Fig. 4. Possible locations of the analogue terminal adaptor.

changes and move to an NGN, while still maintaining continuity of telephone service during the several years of transition. The problem is complex because there is a wide range of PSTN existing business and residential customer types, ranging from those who are "tech savvy" (i.e., computer literate) with high-speed broadband to customers without any computers, broadband access, or even mobile phones. This complexity is captured in Fig. 3, which shows a generalized view of the typical starting point for a telco and the target network. PSTN customers have their telephone service provided from the local serving exchange over landlines, usually composed of a copper pair. The pair may also support voice and data over ISDN (integrated services digital network), as well as different forms of broadband access (e.g., ADSL) or in a hybrid arrangement with optical fiber (e.g. VDSL and G.Fast). Many business premises will still have their telephony switched on site in a digital ISDN private branch exchanges (ISPBX) and linked to the exchange via primary rate (2 Mbit/s or 1.5 Mbit/s) ISDN typically over optical fiber, often using a mixture of signaling systems. There may also be private circuits or alarm circuits carried over the copper lines and terminating at the exchange MDF (for simplicity not shown in Fig. 3).

The target network is an NGN-type common-services IP platform supporting the data and voice services, as described earlier. Here, the customer's devices on the premises interface to the service hub at the IP level – i.e., IP phones for telephony – directly or via the home Wi-Fi system for transmission over high-speed broadband over optical fiber or microwave radio.

The big question for the operators is how best to manage the mix of customer types during the transition to an all-IP (NGN) target network. Based on the experiences of several countries a set of four categories of migration have been identified [5].

Voluntary: Where a subscriber replaces their analogue telephone instrument with an IP telephone of their own volition. This might be done as a result of an upgrade to IT infrastructure at a business premise or early adoption by a tech-savvy residential customer.

Co-incidental (also known as opportunistic migration): When a subscriber has a FTTP broadband access service installed and the copper line is withdrawn this is inherently a VoIP solution. So, the subscriber is expected to provide an IP phone at the time of fiber installation.

Passive: Where a subscriber keeps their existing analogue telephone and copper line even though the operator transfers line to an all-IP network. In this situation the operator provides a "PSTN emulation" facility to allow the customer to experience normal telephony, and possibly remain unaware of the shift to the all-IP network.

Forced: Where the operator requires the subscriber to replace their analogue phones with an IP phone to continue receiving telephony service. The national regulator is responsible for setting the level of advanced warning for the customers.

Clearly, the main determinant for successful transition is how the introduction of IP phones as replacement for the existing analogues phones is managed. If a customer decides to keep their existing telephones, an analogue-terminal adaptor (ATA) is required somewhere between user and the serving NGN node. This device provides much of the PSTN emulation function to convert analogue speech to digital IP packets. It also provides a SIP-gateway to convert multi-frequency (MF) signaling from the phone's keypad. Finally, the ATA usually provides dial tone and ringing current so giving subscribers a familiar experience when making or receiving a call.

Figure 4 shows the possible locations of the ATA – at the serving NGN edge node, at a street cabinet in the access network or on the customer's premises. The ATA is active equipment which requires constant powering. With the ATA located in the network (cases "a" and "b" in Fig. 4) the customer's operator has to maintain power, otherwise the subscriber is re-

quired to provide power (case "c"). There is a consensus that vulnerable people depending on the telephone need a back-up to power their ATA in the event of a break in public power supply. However, countries differ in the national regulator's requirements and who has responsibility for the providing no-break power supply at the customer's premises.

Though there is no inherent cost difference between circuit switching and packet switching [6], the latter is cheaper when deployed as a single multi-service platform for both voice and data services, and importantly, the interface costs are shifted to the customer equipment. Therefore, the cost of the ATA (in the PSTN-emulation scenario) is a penalty for the operator – and the transition business case needs to reflect this. Operators around the world have taken different approaches to coping with this issue.

5. What Has Happened So Far?

In 2004 British Telecom announced it was planning to be the first national operator to replace its legacy networks with an NGN – named 21st Century Network (21 CN) [7]. The design was based on replacing the existing local switching units at the serving exchange buildings by 21 CN edge nodes comprising multiple line terminating units, known as multi-service access nodes (MSAN) linked via ethernet to the IP NGN core network. Crucially, the ATA was housed in the MSAN within the exchange building – i.e., PSTN emulation type "a" in Fig. 4. However, following early field trials, BT decided that moving ahead with this approach was too complex for the technology then available and that the 21 CN program would instead concentrate on providing cost-effective broadband access, since the PSTN exchanges were still working well.

This delay resulted in several benefits. The emerging trend for subscribers to become mobile-only households has reduced the number of ATAs required in the new NGN design. With increased numbers of broadband lines, the number of IP phones owned by subscribers also increased (enabling the voluntary approach described above). Of course, the delay of some 10 years also meant that better technology choices were available for a PSTN replacement. BT's current move to an "all-IP network", is based on PSTN emulation with the ATA located at the line termination on the sub's premises, i.e. type "c".

Interestingly, the independent island territory of Jersey led the move to an NGN with their ambitious program of totally replacing all local copper by optical fiber, together with closure of their PSTN – achieving total conversion by 2018. Their design initially involved a soft-switch-type VoIP exchange replacement, then later moved to an IMS-based NGN approach. The ATA at the at the customer's premises is provided by the operator Jersey Telecom (JT) – i.e., type"c". This approach enabled a common design of integrated IP router and ATA to be deployed at every household irrespective of the type of phone, giving economies of scale in equipment costs and simpler installation processes.

Deutsche Telekom (DT) began their German PSTN transformation program in 2014 having had experience of converting their smaller networks in seven other European countries, particularly Croatia and Slovakia. Their policy was that existing broadband data customers were required to provide an ATA with their router on the premises. However, PSTN emulation was provided to telephony-only subscribers, using MSAN in the serving network node – i.e., type "a". The conversion was substantially completed in 2019, following some important technical upgrades to the program including introduction of IMS and the move to type "c" location of the ATA. In 2020, DT announced that they would be introducing a next-generation IMS platform with network function virtualization with the aim of the "cloudification of voice telephony" using data centers across Germany [8].

Many other countries are following similar approaches in their transformation to all-IP, including Switzerland (completed in 2019) and New Zealand (due to complete in 2022). In contrast to most countries who provide PSTN emulation to support existing analogue voice phones, France is requiring all users to provide an IP phone at the time of conversion. However, the transition period is longer, and subscribers are being given plenty of time to prepare. The target date is 2030 for all operators in France.

It seems that in all cases, the operators are converting their networks on a region-by-region basis, with localized publicity and deployment of installers tightly focused. There are differences in the requirements of the national regulators concerning who should bear the various costs of conversion – for end users and interconnections between operators – as well as the provision of back-up battery powering for vulnerable customers.

Table 1 presents a brief summary of the different rates of progress towards PSTN closure in a range of countries.

6. Cyber and Network Security Issues

Now, to consider the cyber and network security issues of moving the PSTN traffic onto an operator's IP core network, so that telephony is now mixed with all forms of data – latency tolerant and latency intolerant – including video. So, as Fig. 3 illustrates after closure of the PSTN the voice traffic, which enjoyed the protection of the essentially separate PSTN (walled garden of digital circuit switching, SS7 signaling, and dedicated capacity), is thrown onto the common IP platform. Given the headline in a major UK newspaper this September which said: "BT identifying 2,000 signals a second (on their IP core network) indicate cyber-attacks"[1] questions about the wisdom of this move may need to be asked.

A further recent development for the mobile digital IP core is the opening of the network resources to third-parties through a set of APIs. The GSMA launched the Open Gateway initiative at the Mobile World Conference (MWC) 2023. This has the support of 21 global network operators and was the major theme of MWC 2024. Primarily specified for application de-

[1] Guardian newspaper, 13th September 2024

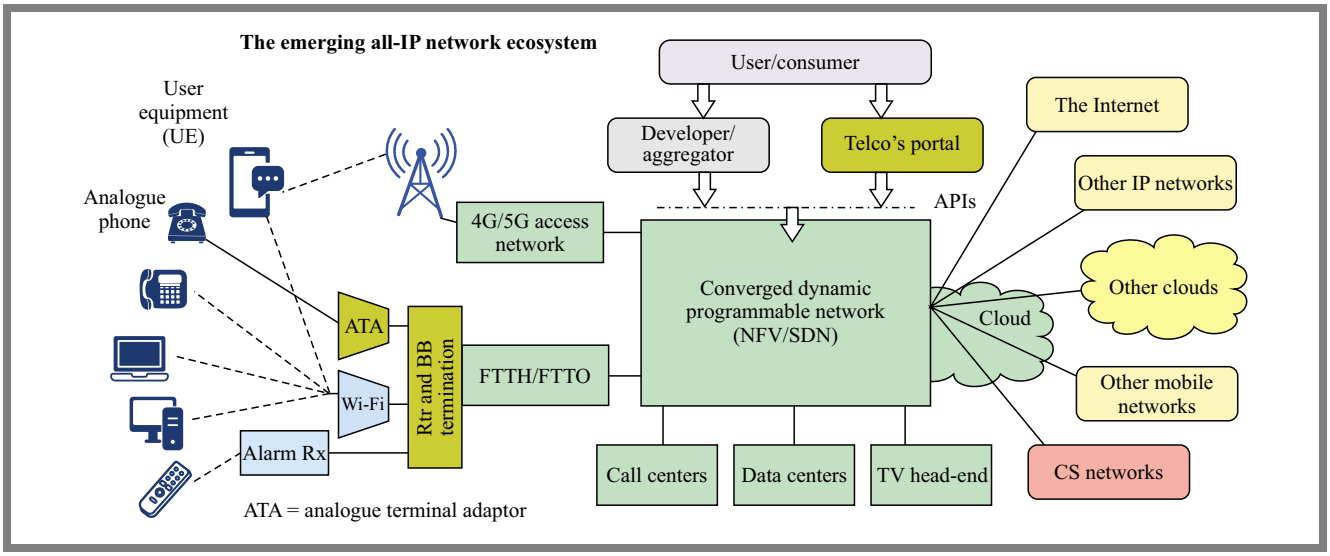


Fig. 5. The all-IP network ecosystem showing integrated fixed-mobile core with network API access.

Tab. 1. Summary of the progress towards PSTN closure in a range of countries.

Country	PSTN switch-off status	Notes
France	Target of 2030 for all operators in France	Closely following the copper network withdrawals
Germany	Substantially completed by 2019	Hampered by business customers not being ready
Italy	Gradual switch-off following fiber coverage towards 2030	
The Netherlands	Recently completed	
Norway	Completed by 2022	
Portugal	About 60% completed, linked to copper withdrawal program	
Spain	Telefonica substantially completed by 2024	
Sweden	PSTN switch off completed by 2010, now 90% of copper network closed	
UK	Both BT and VirginMediaO2 are aiming to substantially complete by 2027	Target of 2025 hampered by alarm systems and customer apparatus problems
Australia	Aims to be substantially closed by 2025, in line with move to National Broadband Network (NBN)	
Japan	Completed 2023	
New Zealand	Aiming for 2030	
Singapore	Completed in 2020	
USA	No national target date. Operators following their own plan, usually following fiber rollout	

velopers, once an application is initiated customers will, in a controlled way, have access to information about other customers location, ID verification, etc.). A wide range of feature and services will be covered by the suite of specified APIs.

Finally, the general direction of network development is towards a single IP core within a country to support both fixed and mobile access networks, with many of the control and transport functions being virtualized and transferred to the cloud, as illustrated in Fig. 5.

7. Author's Conclusions

The aim of this article is to pose some questions and potential concerns about the effect of closing the PSTN and transferring

fixed-line telephony traffic to the emerging common IP core. I hope that these points will get taken into consideration in future cyber security work.

Acknowledgments

This article is an updated and expanded version of one published in the ITP Journal, vol. 16, Part 1, 2022, pp. 9-15. We are grateful to the ITP for permission to publish.

References

[1] A.R. Valdar," Circuit switching evolution to 2012", *The Journal of the Institute of Telecommunications Professionals*, vol. 6, no. 4, 2012.
[2] ITU-T Recommendation Y.2261, *PSTN/ISDN evolution to NGN*, 2006.
[3] IETF Recommendations RFC 2916 & RFC 6116, 2000.
[4] S. Chakraborty, T. Frankkila, J. Peisa, and P. Synnergren, "IMS Multimedia Telephony over Cellular Systems – VoIP Evolution in a Converged Telecommunication World", Wiley, 339 p., 2007 (ISBN: 9780470058558).
[5] G. Forsyth *et al.*, "Preparing the UK for all-IP Future: Experiences from Other Countries", *Broadband Stakeholders Group, Plum Consulting*, 2018 (https://plumconsulting.co.uk/preparing-the-uk-for-an-all-ip-future/).
[6] A.R. Valdar, "Packet versus circuit voice switching", *The Journal of the Institute of Telecommunications Professionals*, vol. 10, no. 1, 2016.
[7] T. Hubbard, "Building the World's Biggest Software-driven Next Generation Network", *Proc. of FITCE Congress*, London, UK, 2008.
[8] M. Kessing, "Voice Telephony from the Cloud", *Deutsche Telekom AG, Media information*, 2020.
[9] "The Future of Fixed Line Telephone Services. Policy Positioning Statement", Ofcom, 2019.

Andy Valdar, Honorary Professor
Department of Electronic & Electrical Engineering
E-mail: a.valdar@ucl.ac.uk
University College London, London, United Kingdom
https://www.ucl.ac.uk/

Privacy-preserving Framework for Automated Detection of Arrhythmia in ECG Data

Kacper Gil and Andres Vejar

AGH University of Krakow, Kraków, Poland

https://doi.org/10.26636/jtit.2025.FITCE2024.2042

Abstract — The integration of machine learning in biomedical engineering applications is crucial to ensure user data security and privacy. This work explores anonymization and differential privacy (DP) frameworks to reduce the risk of biometric identification. The DP method is used to train models in biosignal data without compromising the diagnostic results. The proposed approach for privacy-preserving arrhythmia detection uses a machine learning diagnostic system that reduces discrepancies between prepossessed and raw data, maintaining a correct level of diagnostic precision while improving privacy. The application is evaluated using a control model to analyze the accuracy difference when using privacy-preserving input data.

Keywords — arrhythmia detection, differential privacy, ECG data, privacy enhancing technologies

1. Introduction

Automated diagnostic systems allow reducing the load on health facilities and contribute to improving the quality of medical care at home. Such systems require tracking of several biosignals to monitor the health status of patients. It is important to consider privacy-enhanced methods in the diagnostic system, given that these signals, like electroencephalogram (EEG) and or electrocardiogram (ECG) can reveal the identities of patients using biometric identification methods.

An ideal feature of an automated diagnostic system is the ability to ensure privacy by design [1], [2], where privacy should be built into the technology that supports the system. Important elements to consider during the design phase are the minimization of the user data, the controllability of personal data, the transparency about the system operation, the control on which authorized entities can have data access, and the secure segregation of the data.

1.1. Privacy-enhancing Technologies

For practical engineering implementations, several privacy-enhancing technologies (PET) are available in the literature. Paper [3], specified three general categories: algorithmic PETs, where a formal definition of the algorithms allow to specify strict privacy requirements, architectural PETs, where privacy is enhanced by the design of the underlying distributed computation system, and augmentation PETs, where improvement of the user privacy by the incorporation of generative models of synthetic data and digital twinning is explored. These categories, and relevant examples are presented in Tab. 1. For algorithmic PETs, the most important examples are differential privacy (DP), zero-knowledge proofs, and homomorphic encryption.

Tab. 1. Categories of privacy-enhancing technologies (PETs).

Algorithmic PETs	Architectural PETs	Augmentation PETs
Differential privacy [4]	Federated learning [5]	Synthetic data [6]
Zero-knowledge proofs [7]	Multi-party computation [8]	Digital twinning [9]
Homomophic encryption [10]		

If an external observer cannot verify that the information of a particular user was involved in the computation, then the algorithm is differentially-private. A similar concept concerns zero-knowledge proofs, where two parties, the *verifier* and the *prover* interact to acknowledge the possession of information. The *prover* goal is to acknowledge information possession without disclosing it. Another type of algorithmic PET is homomorphic encryption, it considers the ability of a cryptosystem to perform computations in encrypted data. Upon decryption, it yields an output that is exactly the same as if the operations had been carried out on the unencrypted data.

For architectural PETs, federated learning consist on a distributed strategy, where machine learning models are trained locally, and only the parameters of the models are communicated between the federated peers. Multi party-computation refers to the use of private data in protected computation tasks. All the parties can have access the computed results, but the computation will not reveal the individual data to the peers.

The last type of PETs, refers to augmentation. Synthetic data is data that was created to support and test algorithms and mathematical models, it is specially important in data science and machine learning tasks. A specific type of augmentation is the digital twinning, where a virtual counterpart of a physical system is created to study the real system and to predict its

This work is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) License.
For more information, see https://creativecommons.org/licenses/by/4.0/

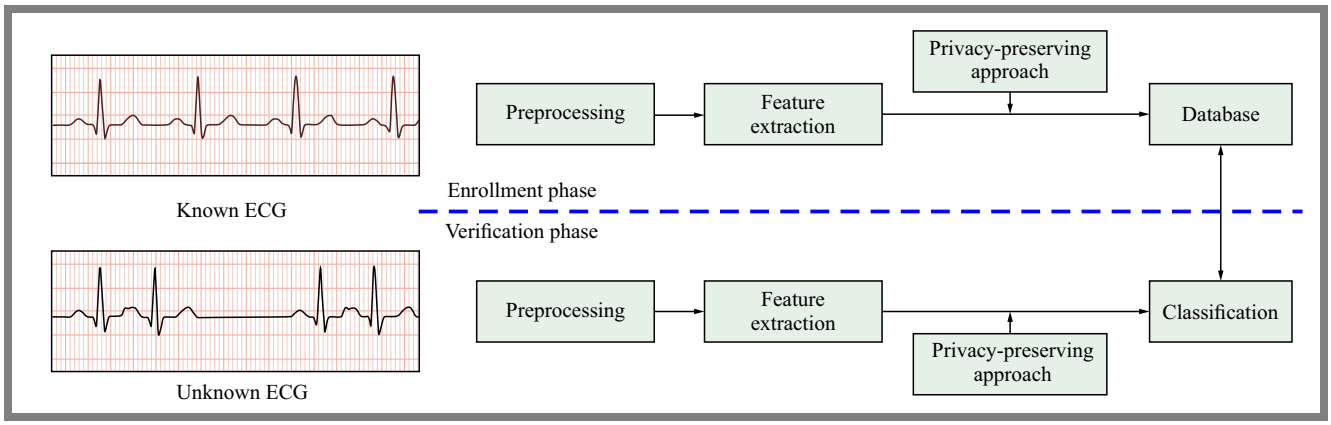


Fig. 1. Biometric identification and re-identification using ECG signals.

response given artificial stimuli. For example, digital twins can be customized to study the evolution of a medical therapy and to predict future results. Anonymized twins can be used by third party healthcare contractors in order to understand the problem under study and to propose privacy-preserving systems to the healthcare facilities.

This work focuses on algorithmic PETs, by the use of DP methods in automated detection of arrhythmia.

1.2. Biometric Identification

There are two phases to the process of biometric identification, namely the enrollment and the verification phases. The enrollment phase is the process of registering a source of biometric data jointly with its associated identification index, with the possibility of including other diverse biometric data, e.g. fingerprints and face image. The data stored are generally processed to obtain a set of features that are characteristic to one person, the biometric template data. The verification phase consists of matching the template data into new data. This phase can be challenging, because biometric data can vary from measurement to measurement.

Biometric identification and authentication using ECG [11]–[13] can be achieved directly or in conjunction with other sources of biometric data. It is interesting because it can be used as a continuous authentication method in critical systems, for example in continuous driver authentication for cash transport, public transportation, military, and car rental and sharing services [14].

The working principle of biometric identification of ECG in a diagnostic system can be seen in the Fig. 1. The enrollment phase consist on the preprocessing of the *Known ECG* signal, i.e. creating a pair (ECG signal, user ID). In order to compare the ECG signal with another *Unknown ECG* signal it is required to extract features of the signal. These features will be stored as a template in the system database. A privacy-preserving approach for registering the ECG features in the database will enhance the security diagnostic service. For example, only enrolled users will be able to be classified by the arrhythmia detection service. In this work, we focus in implementing a differentially-private classification model for ECG diagnostic, guarantying that no database sample can significantly affect the outcome of the classification.

1.3. Medical Background of Arrhythmia

Arrhythmia is a medical condition characterized by an irregular heartbeat, also classified as tachycardia or bradycardia if the heart beats too fast or too slow, respectively. Alternatively, the irregularity can display no pattern; in such cases it is called fibrillation. Factors of increased risk of arrhythmia include cardiovascular disease, heart surgery, and cardiomyopathy that implies changes in heart structure. Other causes not related to the heart are electrolyte imbalances, medications, and certain stimulants. Personal lifestyle also plays a role in the incidence rate of heart irregularities. High levels of stress, smoking, and physical exertion are the most common.

Generally, arrhythmias manifest only as palpitations, light dizziness, and shortness of breath. However, in more severe cases it can lead to fainting and may even be life-threatening. The diagnosis procedure involves the use of an ECG, usually taken over a period of 24–48 h, with the help of a Holter monitor.

Several arrhythmia detection methods [15]–[17], can be found in the literature, where the Physionet computing in cardiology challenge and its ECG dataset is an important benchmark for machine learning methods [18]. More advanced data sets are also available, for example, the 12 leads ECG data set [19].

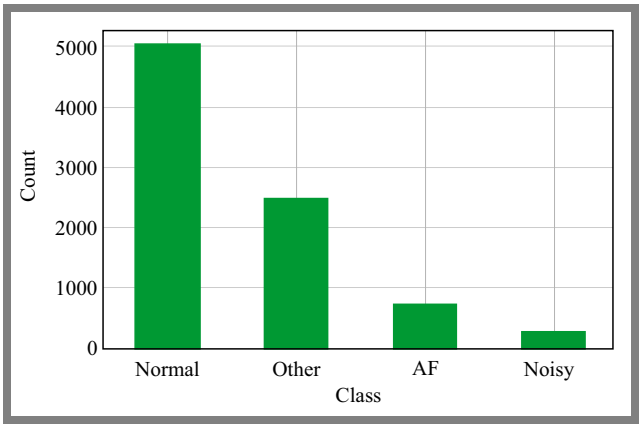


Fig. 2. Distribution of target classes in the dataset.

Therefore, there are a growing collection of automated methods of arrhythmia detection and classification [20] that can benefit greatly with the incorporation of PETs.

2. Materials and Methods

2.1. Data Sourcing and Labeling

The study from [21] proposes feature-based classifiers and convolutional neural network (CNN) models for arrhythmia classification using the first minute of each ECG sample. In this work, we use a similar approach considering a CNN model, but we generate 2D images of the ECG samples using recurrence plots [22]. Another important difference is that we restrict the length of input data for the CNN classifier to a very short time period of around 2.3 s. To train and to test the model, we use random sampling to consider any period of 2.3 s in the full length timeseries. Our model is designed for real time detection using the buffer of 2.3 s.

The data was provided by AliveCor for the purposes of the aforementioned challenge [18]. The total number of ECG recordings exceeded 12 000. Each of them was taken using single-channel ECG devices of one of three generations. The electrodes, mostly, were placed in each hand of a patient, resulting in lead I (LA-RA) ECG. Many of the data series were inverted, creating (RA-LA) series. The signal recordings average at around 30 s. The equipment then transmitted the data to a portable device over radio waves using 19 kHz carrier frequency and a modulation index of 200 Hz/mV. The data was digitized in 16-bit files with a sample frequency of 300 Hz.

The experts have divided the data into 4 classes: 0 – normal rhythm, 1 – atrial fibrillation (AF) rhythm, 2 – other (abnormal) rhythm, and 3 – noisy recording. The distribution of the classes can be seen in the Fig. 2.

2.2. System Description

In Fig. 3 the diagram of the proposed approach to preserving the privacy of the arrhythmia detection system is presented. This research examines a machine learning diagnostic system in which raw ECG biosignals x undergo client-side preprocessing to become a filtered signal u. Subsequently, this signal is utilized by the diagnostic system g at the diagnostic center. The goal of this system is to reduce the discrepancy between the results of the preprocessed g and a raw data classifier f, $f(x) \approx g(u)$, thus maintaining high diagnostic precision while improving privacy. The application is tested with the control model f that is not privacy preserving, to compare the accuracy level of the arrhythmia detection.

The use of recurrence plots and phase space analyses have seen use in some classification approaches [23], [24].

Out of the input data 700-samples-long snippets (2.3 s) were randomly cropped and later transformed into image data using the `RecurrencePlot` function from the `pyts` library [25]. The threshold and percentage values were set to "point" and 20 respectively. These parameters are used for binarization of the recurrence plot, that consist in a 700×700 pixel image with a single color channel. The images were resized using bilinear. interpolation to 350×350 pixels.

This data was then shuffled and passed to the CNN model consisting in three 2D-convolution layers for acquiring image features and three dense layers acting as the output classifier. The total number of trainable parameters of the CNN model is 9 539 669.

In this application we aggregate the target classes into two: 0 – normal rhythm and 1 – atrial fibrillation (AF) rhythm or other (abnormal) rhythm. Noisy recordings are excluded from the dataset given that, this is a problem that needs to be addressed early, during signal acquisition [26], [27].

The privacy levels are controlled by the parameters for $\varepsilon > 0$ and $\delta \in [0, 1)$.

The classifier g, that takes an input u and returns the output y, is (ε, δ)-differentially-private for two similar datasets U_a and U_b, $U_a \cap U_b \neq \emptyset$ if the following relation is established:

$$\mathbf{P}\left(g(u_a \in U_a)\right) \leqslant \exp(\varepsilon)\, \mathbf{P}\left(g(u_b \in U_b)\right) + \delta\,. \qquad (1)$$

Smaller choices of the parameter ε make the model more private, controlling the level of noise. The parameter δ refers to the probability of a data breach. It is pertinent to set ε and δ to achieve a trade-off between privacy and classification performance.

3. Results

In order to compare a privacy-unaware classification model of arrythmias with a differentially private model, we trained a baseline model and a differentially-private model. Both models share the same CNN architecture. The DP training of the models was performed using the Opacus library [28]. The models were developed using the PyTorch library for deep learning in a GPU NVIDIA GeForce RTX 4080.

3.1. Baseline CNN Classification

The initial output of the CNN classification model, constructed to explore the performance of DP models, can be seen in the Fig. 4.

One important application of online detection and diagnostic systems is to trigger alarms or alert the corresponding medical services in case of an improper heart rhythm. Consequently, the most important factor to minimize is the number of false negatives in the classification. We use the false omission

Tab. 2. Model metrics.

Metric	Initial model	Final model
FOR	0.3982	0.1274
Accuracy	0.7846	0.8846
Precision	0.8608	0.9029
Sensitivity	0.6018	0.8230
Specificity	0.9252	0.9328
F1 score	0.7083	0.8611

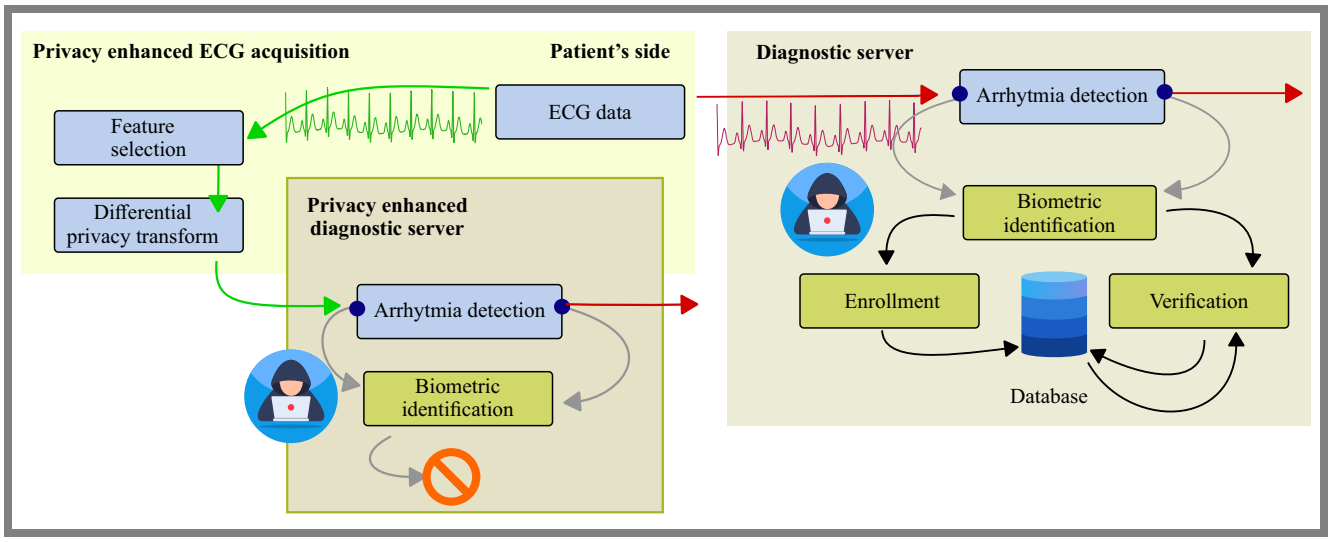


Fig. 3. System diagram considering raw and privacy-enhanced arrhytmia detection. In the left side is presented the proposed privacy enhanced arrhytmia diagnostic system. A compromised, raw ECG arrhytmia diagnostic system is depicted in the right side.

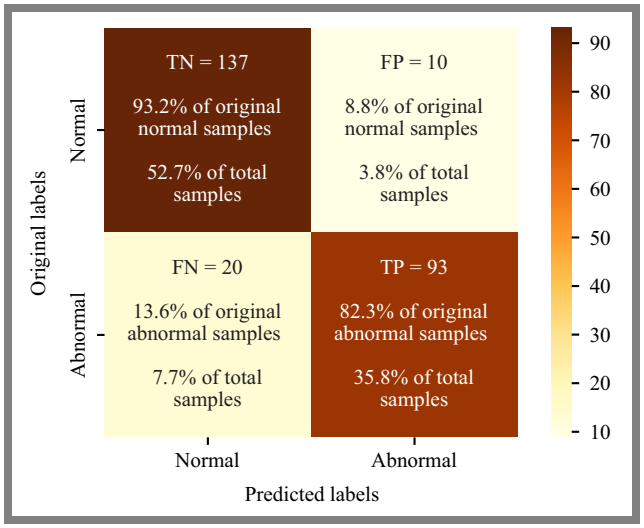


Fig. 4. Confusion matrix of the baseline model.

rate (FOR), to measure the proportion of incorrect negative clasifications, false negatives (FN) with respect to the overall negative class:

$$\text{FOR} = \frac{\text{FN}}{\text{TN} + \text{FN}} . \tag{2}$$

The selected metrics for comparison of the baseline model in the initial and final versions are presented in Tab. 2.

The metrics presented in Tab. 2 show some notable improvements after optimization, including a reduction in the false omission rate (from 0.3982 to 0.1274), an increase in sensitivity (from 0.6018 to 0.8230), and an improved F1 score (from 0.7083 to 0.8611). These improvements suggest that the optimization efforts have enhanced the classifier's ability to detect cardiac arrhythmias, particularly in minimizing missed detections. Said parameter is of main concern as it can be feasibly presumed that the potential user will already have a history of prior medical issues with heart rhythm. That is, a possible false alarm will not be as damaging as a missed anomaly, given that the user will have a way of turning it off.

The remaining parameters also saw an increase, i.e. precision (from 0.8608 to 0.9029) and specificity (from 0.9252 to 0.9328), which points to the overall improvement of the model.

The final accuracy of 88.46% is a considerable improvement over the initial 78.46%. The model was designed to process short ECG signals as inputs (about 2.3 s), which makes it particularly useful in real-world scenarios where rapid detection is critical. Furthermore, the model is currently in its preliminary stages of development, and future iterations are planned to incorporate greater code complexity, which is expected to further improve performance across all metrics.

3.2. DP Implementation in the CNN

To compare the selection of privacy hyper-parameters with respect to the performance in retraining, firstly a DP version of CNN classifier studied in the previous subsection was trained from scratch with weak privacy parameters until achieving a classification accuracy of 75%. Said threshold was preselected for testing the capabilities of DP implementation in ECG classification. It can be potentially improved with training-time optimization techniques.

Secondly, the classifier was loaded and retrained with different choices of the privacy hyper-parameters:

- Maximum gradient norm G – corresponds to the maximum achievable norm for each gradient sample. Greater gradients will be clipped to the value of this parameter. With higher values of the maximum gradient norm higher levels of privacy are achieved.
- Privacy budget P_ε – cumulative value of the ε parameter over all epochs during training. With smaller ε values, higher levels of privacy are achieved.
- δ – the likelihood of a data breach.

For all the experiments, δ was fixed in 1.1, the values of P_ε include 12, 48, 84, and 120. The results are presented in Tab. 3 according to maximum gradient norm. Training stage con-

sisted of 20 epochs. The preliminary DP implementation was able to achieve 75% accuracy. Taking that into consideration, it can be observed that higher levels of G values substantially affected the model performance.

4. Conclusions and Further Work

In this work, a privacy-preserving framework for the detection of arrhytmia is presented. The framework considers a *privacy enhanced ECG acquisition* on the patient's side, useful for remote diagnostic in homecare, and a *privacy enhanced diagnostic server* that provides the automated diagnostic service.

The ongoing work considers a validation of the results with standard ECG biosignal databases. It is important to evaluate the performance penalty of implementing DP, or other PETs in standard deep learning models. One of the objectives of this work is to promote the application of privacy-enhancing technologies in the early stages of automated diagnostic systems, revisiting well proven classification methods and incorporating privacy-enhancing hyper-parameters during design and learning phases.

Further work will consider the design of automated diagnostic systems with the joint goal of security and privacy. In addition, the use of a large set of sensors (e.g. temperature, pulse oximetry, EEG, and EMG) and different target diseases for detection can be explored as an extension of the proposed approach.

Tab. 3. Performance after DP retraining.

G	P_ε	Train loss	Train acc.	Test loss	Test acc.
1.10	120	1.03	73.35	1.01	72.99
1.10	84	1.03	73.37	1.00	73.00
1.10	48	1.04	73.23	1.01	73.19
1.10	12	1.06	72.89	1.02	72.45
4.07	120	0.91	50.70	0.70	49.57
4.07	84	0.97	50.33	0.70	49.51
4.07	48	0.83	50.02	0.70	49.56
4.07	12	4.45	50.55	1.06	51.30
7.03	120	2.25	50.31	0.79	51.18
7.03	84	4.99	50.74	3.76	51.52
7.03	48	21.45	50.35	16.60	51.03
7.03	12	65.63	50.62	188.09	50.65
10.00	120	21.18	50.24	3.48	50.40
10.00	84	45.87	50.80	0.98	50.58
10.00	48	7.30	50.29	1.02	51.54
10.00	12	168.60	50.35	50.34	50.05

Acknowledgments

This research was supported by the National Research Institute, grant number POIR.04.02.00-00-D008/20-01, on "National Laboratory for Advanced 5G Research" (acronym PL-5G) as part of the Measure 4.2 Development of modern research infrastructure of the science sector 2014–2020 financed by the European Regional Development Fund.

References

[1] P. Schaar, "Privacy by Design", *Identity in the Information Society*, vol. 3, pp. 267–274, 2010 (https://doi.org/10.1007/s12394-010-0055-x).

[2] A. Nordgren, "Privacy by Design in Personal Health Monitoring", *Health Care Analysis*, vol. 23, pp. 148–164, 2013 (https://doi.org/10.1007/s10728-013-0262-3).

[3] S. Jordan, C. Fontaine, and R. Hendricks-Sturrup, "Selecting Privacy-enhancing Technologies for Managing Health Data Use", Frontiers in Public Health, vol. 10, 2022 (https://doi.org/10.3389/fpubh.2022.814163).

[4] M. Yang *et al.*, "Local Differential Privacy and its Applications: A Comprehensive Survey", *Computer Standards & Interfaces*, vol. 89, art. no. 103827, 2023 (https://doi.org/10.1016/j.csi.2023.103827).

[5] K.K. Coelho *et al.*, "A Survey on Federated Learning for Security and Privacy in Healthcare Applications", *Computer Communications*, vol. 207, pp. 113–127, 2023 (https://doi.org/10.1016/j.comcom.2023.05.012).

[6] M. Giuffrè and D.L. Shung, "Harnessing the power of synthetic data in healthcare: innovation, application, and privacy", *Digital Medicine*, vol. 6, art. no. 186, 2023 (https://doi.org/10.1038/s41746-023-00927-3).

[7] L. Petrosino *et al.*, "A Zero-knowledge Proof Federated Learning on DLT for Healthcare Data", *Journal of Parallel and Distributed Computing*, vol. 196, art. no. 104992, 2024 (https://doi.org/10.1016/j.jpdc.2024.104992).

[8] T. Liu, "Research on Privacy Techniques Based on Multi-Party Secure Computation", *2024 3rd International Conference on Artificial Intelligence and Autonomous Robot Systems (AIARS)*, Bristol, United Kingdom, 2024 (https://doi.org/10.1109/AIARS63200.2024.00171).

[9] C.S. Jørgensen, A. Shukla, and B. Katt, "Digital Twins in Healthcare: Security, Privacy, Trust and Safety Challenges", *Proc. of European Symposium on Research in Computer Security*, pp. 140–153, 2023 (https://doi.org/10.1007/978-3-031-54129-2_9).

[10] K. Munjal and R. Bhatia, "A Systematic Review of Homomorphic Encryption and its Contributions in Healthcare Industry", *Complex & Intelligent Systems*, vol. 9, pp. 3759–3786, 2023 (https://doi.org/10.1007/s40747-022-00756-z).

[11] A.D.C. Chan, M.M. Hamdy, A. Badre, and V. Badee, "Person Identification using Electrocardiograms", *2006 Canadian Conference on Electrical and Computer Engineering*, Ottawa, Canada, 2006 (https://doi.org/10.1109/CCECE.2006.277291).

[12] J. Xu, T. Li, Y. Chen, and W. Chen, "Personal Identification by Convolutional Neural Network with ECG Signal", *2018 International Conference on Information and Communication Technology Convergence (ICTC)*, Jeju, South Korea, 2018 (https://doi.org/10.1109/ICTC.2018.8539632).

[13] S. Asadianfam, M.J. Talebi, and E. Nikougoftar, "ECG-based Authentication Systems: A Comprehensive and Systematic Review", *Multimedia Tools and Applications*, vol. 83, pp. 27647–27701, 2023 (https://doi.org/10.1007/s11042-023-16506-3).

[14] L.D. Chhibbar *et al.*, "Enhancing Security Through Continuous Biometric Authentication Using Wearable Sensors", *Internet of Things*, vol. 28, art. no. 101374, 2024 (https://doi.org/10.1016/j.iot.2024.101374).

[15] A.Y. Hannun *et al.*, "Cardiologist-level Arrhythmia Detection and Classification in Ambulatory Electrocardiograms Using a Deep Neural Network", *Nature Medicine*, vol. 25, pp.65–69, 2019 (https://doi.org/10.1038/s41591-019-0359-9).

[16] R. Li *et al.*, "Interpretability Analysis of Heartbeat Classification Based on Heartbeat Activity's Global Sequence Features and BiLSTM-attention Neural Network", *IEEE Access*, vol. 7, pp. 109870–109883, 2019 (https://doi.org/10.1109/ACCESS.2019.2933473).

[17] S. Mousavi and F. Afghah, "Inter- and Intra-patient ECG Heartbeat Classification for Arrhythmia Detection: A Sequence-to-Sequence Deep Learning Approach", *ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Brighton, United Kingdom, 2019 (https://doi.org/10.1109/ICASSP.2019.8683140).

[18] G.D. Clifford *et al.*, "AF Classification from a Short Single Lead ECG Recording: the PhysioNet Computing in Cardiology Challenge 2017", *2017 Computing in Cardiology Conference (CinC)*, Rennes, France, 2017 (https://doi.org/10.22489/CinC.2017.065-469).

[19] E.A. Perez Alday *et al.*, "Classification of 12-lead ECGs: the PhysioNet/Computing in Cardiology Challenge 2020", *Physiological Measurement*, vol. 41, art. no. 124003, 2020 (https://doi.org/10.1088/1361-6579/abc960).

[20] Y. Ansari, O. Mourad, K. Qaraqe, and E. Serpedin, "Deep Learning for ECG Arrhythmia Detection and Classification: An Overview of Progress for Period 2017–2023", Frontiers in Physiology, vol. 14, art. no. 1246746, 2023 (https://doi.org/10.3389/fphys.2023.1246746).

[21] F. Andreotti *et al.*, "Comparing Feature-based Classifiers and Convolutional Neural Networks to Detect Arrhythmia from Short Segments of ECG", *2017 Computing in Cardiology Conference (CinC)*, Rennes, France, 2017 (https://doi.org/10.22489/CinC.2017.360-239).

[22] B.M. Mathunjwa *et al.*, "ECG arrhythmia classification by using a recurrence plot and convolutional neural network", *Biomedical Signal Processing and Control*, vol. 64, art. no. 102262, 2021 (https://doi.org/10.1016/j.bspc.2020.102262).

[23] S.-C. Fang and H.-L. Chan, "QRS Detection-free Electrocardiogram Biometrics in The Reconstructed Phase Space", *Pattern Recognition Letters*, vol. 34, pp. 595–602, 2013 (https://doi.org/10.1016/j.patrec.2012.11.005).

[24] B. M. Mathunjwa *et al.*, "ECG Recurrence Plot-based Arrhythmia Classification Using Two-dimensional Deep Residual CNN Features", *Sensors*, vol. 22, art. no. 1660, 2022 (https://doi.org/10.3390/s22041660).

[25] J. Faouzi and H. Janati, "pyts: A Python Package for Time Series Classification", *Journal of Machine Learning Research*, vol. 21, 2020 (https://jmlr.csail.mit.edu/papers/volume21/19-763/19-763.pdf).

[26] S.E. Mathe, N.K. Penjarla, S. Vappangi, and H.K. Kondaveeti. "Advancements in Noise Reduction Techniques in ECG Signals: A Review", *2024 IEEE 3rd World Conference on Applied Intelligence and Computing (AIC)*, Gwalior, India, 2024 (https://doi.org/10.1109/AIC61668.2024.10730852).

[27] F. Liu, Y. Xu, and Y. Yao, "Highly Efficient Low Noise Solutions in ECG Signals", *Journal of Physics: Conference Series*, vol. 2246, art. no. 012030, 2022 (https://doi.org/10.1088/1742-6596/2246/1/012030).

[28] A. Yousefpour *et al.*, "Opacus: User-friendly Differential Privacy Library in PyTorch", *ArXiv*, 2021 (https://doi.org/10.48550/arXiv.2109.12298).

Kacper Gil, B.Eng.
Institute of Telecommunications
https://orcid.org/0009-0002-9404-9720
E-mail: kagil@student.agh.edu.pl
AGH University of Krakow, Kraków, Poland
https://www.agh.edu.pl

Andres Vejar, Ph.D.
Institute of Telecommunications
https://orcid.org/0000-0002-2041-0387
E-mail: avejar@agh.edu.pl
AGH University of Krakow, Kraków, Poland
https://www.agh.edu.pl

Enhancing DGA Detection with Machine Learning Algorithms

Hubert Biros and Mirosław Kantor

AGH University of Krakow, Kraków, Poland

https://doi.org/10.26636/jtit.2025.FITCE2024.2033

Abstract — The domain generation algorithm (DGA) is a popular technique used by malware to reliably establish a connection to a command and control (C&C) server. Pseudo-random domain names generated by DGA are used to bypass security measures and allow attackers to maintain control over malware-infected devices. In this work, we present a two-pronged approach to detecting character-based and word-based DGA domain names, creating classifiers specifically tailored to each type. For character-based DGA detection, we employed seven traditional machine learning methods: support vector machine, extremely randomized trees, logistic regression, Gaussian naive Bayes, nearest centroid, random forests, and k-nearest neighbors. We applied a featureful approach, using features extracted from the domain names themselves. Some of these features were drawn from existing literature, while others were newly proposed by authors. Feature selection techniques were used to retain only the best-performing ones. For the more complex task of detecting word-based DGA domain names, we used CNN and LSTM models, relying solely on word embeddings derived from the domain name components. Performance evaluation shows that proposed method gives high-performing, specialized DGA classifiers, which can be combined to create a more general-purpose classifier.

Keywords — character-based DGA, cybersecurity, DGA detection, DNS, machine learning-based DGA detection, malware, word-based DGA

1. Introduction

The domain name system (DNS) is a critical part of Internet infrastructure, translating human-readable domain names into machine-readable IP addresses. As the Internet evolves, securing the DNS against emerging threats becomes increasingly challenging. One common threat is the abuse of DNS through domain generation algorithms (DGAs), which malware uses to bypass security measures.

Devices infected by the malware, such as botnets or ransomware, need a reliable way to establish a connection with the command and control server (C&C) [1]–[3]. C&C plays a key role in operating malware-infected devices, allowing attackers to control victim machines and extract from them sensitive and valuable data [1]. Infected devices need a way to get the address of their C&C servers. Hard-coding the IP addresses or the domain names of these in the malware source code, is not a good solution, since once those are found by some security intelligence, blacklists can be created to shut down the operation of the malware [4], [5]. Instead, some technique must be used by malware creators in order to easily relocate the C&C server to a different location in case of take-down of the working C&C server [3].

DGA is a popular technique used to establish a communication channel between infected devices and C&C servers [5]. For instance, many of the top 10 most popular financial malware families in the year 2023 were employed with some variant of DGA [6]. DGA is basically a piece of code that generates a large number of pseudo-random domain names that infected devices try to resolve to the address of the C&C server. In order to generate different sets of domains every time period, DGA typically uses some kind of seed in the form of a numerical hard-coded value or some time-dependent number [7], [8]. The key idea behind DGA is that malware operators having the same DGA algorithm and seed can register some of the generated domains and allow infected machines to connect with C&C server [8].

The process of using DGA is illustrated in Fig. 1, where the hacker or person who is put in charge of the infected devices uses the DGA algorithm implemented in the malware along with the particular seed to generate a set of domain names from which he selects one to register in the global DNS. In this case, the domain name is knosszts.ru. The hacker knows that in the future the infected device will use the same seed and thus generate the same set of domain names. The device will try to resolve all of them until one of them is correctly resolved and points to the C&C server.

This paper provides new insights into the detection of DGA domain names. Specifically, we introduce four new features that enhance the detection of character-based DGAs. Feature selection techniques revealed that some features used in previous works may be unnecessary and can be omitted without impacting performance. Additionally, our approach of using word embeddings for detecting word-based DGAs has shown very promising results. The strong performance of these DGA-focused classifiers suggests that they could serve as a solid foundation for future research and improvements in this area.

This paper is organized as follows. Section 2 introduces readers to the different types of DGAs and presents the approaches used to detect the domains generated by these algorithms. This section will additionally provide an overview of the related papers in the field and present the state-of-the-art solutions that were used for comparison with the results of our work. Section 3 provides a brief introduction to the

This work is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) License.
For more information, see https://creativecommons.org/licenses/by/4.0/

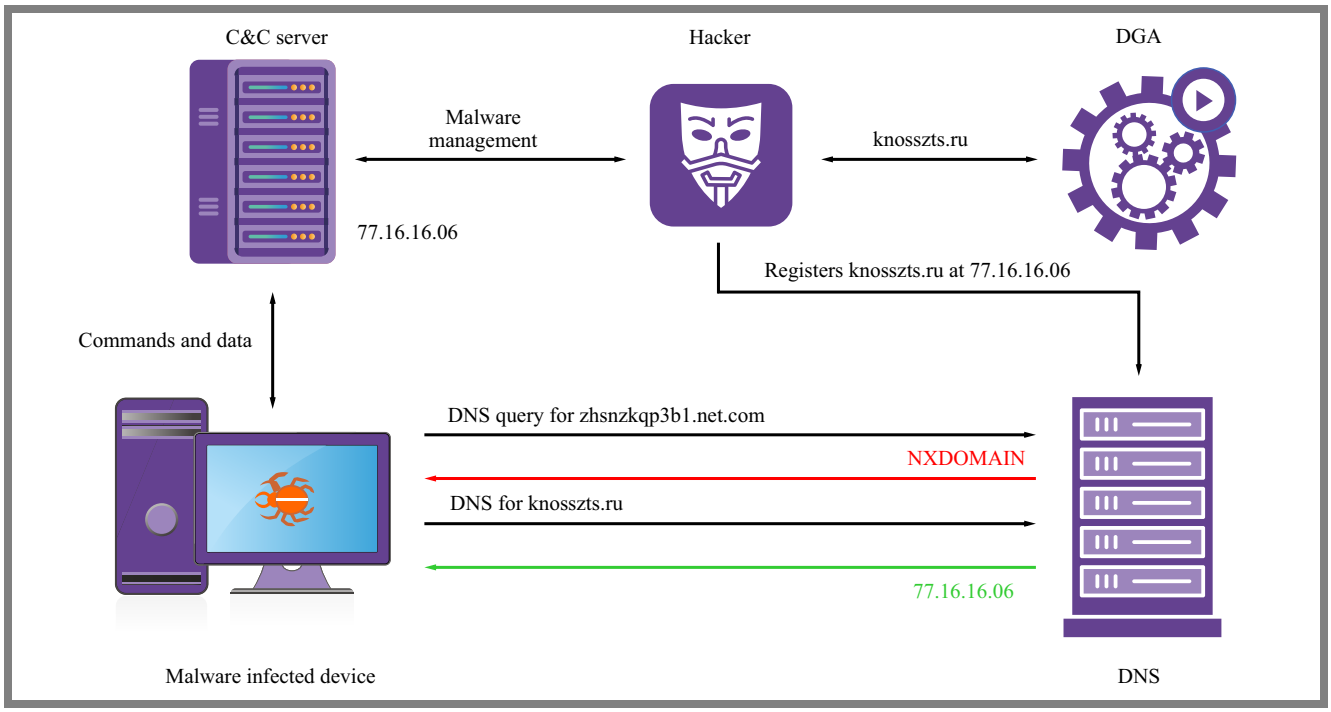


Fig. 1. Malware-infected device using DGA to generate domain names and contact C&C server.

machine learning methods that were used to create the DGA classifiers. Sections 4 and 5 provide a detailed description of the created models for character-based and word-based DGA detection, respectively. The evaluation results of the obtained DGA classifiers can be found in Section 6. The conclusions are presented in Section 7.

2. Related Works

This section provides a comprehensive analysis of domain generation algorithms. It offers insights into the intricacies of DGAs, their mechanisms, and the challenges of detecting them. Following subsections offers an overview of existing research in DGA detection along with the various approaches used in the field and briefly describes other works that were used for comparison with proposed models. A full comparison of the evaluation metrics can be found in Tab. 6.

2.1. Overview of DGA Types

We can distinguish three main categories of DGAs: character-based DGAs, word-based DGAs (sometimes called dictionary DGAs), and mixed DGAs, which combine elements of the first two types [8], [9]. Though another classification scheme, as outlined in [10], identifies four types: arithmetic-based DGAs, hash-based DGAs, wordlist-based DGAs, and permutation-based DGAs, we will adhere to the former classification. This is because it groups domain names based on how they appear to a human observer, which aligns better with the focus of our study.

The character-based DGA domains are constructed, by concatenating random characters into strings and then adding top level domain (TLD) to form the domain name. *Conficker*, *Necurs*, or *Cryptolocker* are examples of malware that use this type of DGA. Some character-based DGAs are slightly more sophisticated, and in the process of creating the domain names, they distinguish between vowels and consonants to make generated domains more pronounceable. Examples of malware that uses this type of DGA are *Pitou* and *Symmi*. Word-based DGA are more dangerous in the sense they are more difficult to detect even by humans since they use a pre-defined list of words in the process of generating the domain names.

Word-based DGA can distinguish between different parts of speech such as adjectives, nouns, or verbs in order to create even more benign-looking domains. Examples of malware using word-based DGA are *Matsnu*, *Rovnix*, or *Suppobox*.

The last category is mixed DGA which is a combination of the two previous categories. Domains generated by mixed DGA have one part of the domain name generated by character-based DGA and another part is some combination of words. An example of malware that uses this category is *Banjori*. Table 1 contains examples of domains generated by different DGA-based malware families.

2.2. DGA Detection Approaches

The methods of detecting DGA domains can be divided into 2 categories: inline and retrospective [8], [11], [12]. In the retrospective method, collected DNS traffic can be analyzed in order to find DGA domains. This approach is a typical example of the intrusion detection system (IDS), since it works on past DNS data and cannot be used to stop malware from its operation. The latter inline approach can detect DGA domains as soon as the DNS query is made.

Tab. 1. Samples of different DGA domain names (C – character-based DGA, W – word-based DGA, M – mixed DGA).

Malware family	DGA type	Sample domain name
Banjori	W	uotvestnessbiophysicalohax.com
Cryptolocker	C	xpbfsnbuabrne.co.uk
Pitou	C	xoaomasat.us
Matsnu	W	mirrorhusbandboxconflict.com
Suppobox	W	weatheranother.net

The methods of detecting DGA domains can be further divided based on the information they need to classify the domain as DGA or non-DGA. The simplest approach is to use the information contained in the domain name string itself. While this method can effectively create a DGA classifier for character-based domains, such as "x1df6f33a99a10f9c7fdc5d176cd405ed7.so" generated by *Dyre* malware, word-based DGA domains, like "emilsmusic.com" generated by *Emotet*, may often appear more benign and less indicative of artificial origin. Although they can still be detected using their domain name, incorporating additional side information as parameters from DNS traffic (e.g., TTL, IP addresses) or information from the WHOIS database can enhance classification accuracy [12], [13].

The state-of-the-art solutions for detecting DGA domain names are based on machine learning methods to create DGA classifiers. Machine learning-based classifiers can be created in two ways. One way is the featureful approach in which we create a set of features from a domain name or other side information and use it to design the detection model. Examples of works that follow this direction include: [4], [9], [14], [15].

Another method used by [16]–[18] is the featureless approach in which we use some deep learning techniques like neural, convolutional, or LSTM networks to create classifiers. In this approach, the DGA classifier works on the information directly contained in the domain name and no features need to be constructed based on it. Some works like [8], [13] present a combined approach in which the classifier is trained using both featureless and featureful manner.

In detecting DGA domains, many works use n-gram analysis, e.g., [4], [14], [19]–[21]. N-grams are sequences of n consecutive characters in a string. For example, we can extract the following list of 2-grams from the label "google": ["go", "oo", "og", "gl", "le"]. There are methods for detecting DGA domains only by using statistics of n-gram distribution like [21], which presents a model for detecting DGA domains based on a reputation score calculated by segmenting the domain names into n-grams and then calculating a weight value for each resulting n-gram based on the occurrence of the corresponding n-gram in non-DGA domains.

2.3. Works Used for Comparison with Proposed Models

Many DGA detection proposals focus on proposing a universal classifier capable of detecting both character-based and word-based DGA domains. However, as highlighted in the findings of [4], some such models only work well in detecting only a specific type of DGA. We reviewed several state-of-the-art models in the literature that are either strictly designed to detect one type of DGA or have been trained using only one type of DGA.

The performance metrics of these models, as reported in the cited papers, should not be directly compared with our models since they were evaluated using different datasets, consisting of varying benign DNS traffic and distinct sets of DGAs. However, we include these in Tab. 4 to provide a broader perspective on the landscape of DGA detection models. Table 4 also shows some other general approaches to detecting DGA domains for a more comprehensive overview. Below is a brief description of these state-of-the-art solutions.

Paper [19] proposed several DGA detection models, the best of which was the random forest classifier achieving an accuracy of 90.80%. The authors used a dataset of 30 000 benign domains and 30 000 character-based DGA domains that are used by four malware families: *Cryptolocker*, *Goz*, *Newgoz*, and *Conficker*, which are all character-based DGAs. Of the character-based DGA detection models we proposed, only two (Gaussian naive Bayes and nearest centroid) proved inferior. The same algorithm used in this work achieves an accuracy of 97.03% while maintaining a lower false positive rate (FPR).

The authors of [14] proposed a random forest classifier that uses 24 classification features extracted from each domain name. The model was trained on a dataset of 200 000 and tested on a set of 53 200 DGA domains of 39 different malware families. The model achieved a good result with 97.03% accuracy but performed poorly in detecting word-based DGA domains. In addition, the classifier was unable to correctly classify any *Banjori* botnet domains that implement mixed DGA. In our case, the random forest model with an extended set of 25 features designed to detect character-based DGA was able to detect 98.88% of Banjori DGA domains.

Article [21] did not use machine learning techniques to build a DGA classifier, but instead n-gram analysis of the domain names was proposed. It used 8 000 benign and 2 265 DGA domains belonging to various unspecified DGAs to evaluate the model.

The authors of [22] used only character-based and mixed DGA-generated domains, so it can be directly compared with the seven DGA classifiers proposed in our work. The proposed model is based on the LSTM network and uses a rather large dataset with a total of 1 675 404 domains, 10% of which serves as a test dataset. The authors of the referenced study reported only precision (PPV), true positive rate (TPR), F1-score, and accuracy in their results. Three of our seven character-based DGA domain detection models (KNN, RF, and ET) outperform it, although KNN achieves a slightly lower TPR (94.98% vs. 95.14%).

The paper [23] focused on the detection of word-based DGA domains therefore can be directly compared to the CNN and LSTM models we presented. Domains produced by DGA $Matsnu$ and $Suppobox$ were used in the evaluation of the

proposed models. The best classifier turned out to be random forest, which is outperformed by our two proposed models for word-based DGA detection in terms of ACC value.

Several classifiers for detecting word-based DGA domains was proposed in [9]. The best obtained classifier was the J-48 decision tree and it proved to be slightly better than our LSTM network model, but the CNN network model outperforms it in all available evaluation metrics.

In [15] DGA detection models based on Kullback-Leibler divergence and Jaccard index-based metrics are presented. The best multilayer perceptron (MLP) classifier was tested using 25 different DGAs, including character-based DGAs and word-based DGAs.

In [18] an LSTM model is proposed using both character-based and word-based DGA domain names in the dataset, but with a much smaller share of the latter.

The authors of [24] present a neural network model using BiLSTM and CNN layers with an attention mechanism (ATT-CNN-BiLSTM). The dataset contains 24 different DGA domain names, both character-based and word-based.

Article [17] proposed another approach to detect only the domain names generated by word-based DGAs. The authors present an ensemble learning-based model using both LSTM and CNN networks. The dataset consisted of domain names generated by *Suppobox*, *Gozi*, and *Matsnu* malware. Similar to the paper [9], our proposed LSTM solution is slightly inferior to this ensemble model, but the CNN model outperforms it.

[25] presented a deep neural network (DNN) model for creating a DGA classifier based on features extracted from the domain names and DNS traffic. The paper used 5 different character-based DGAs in the dataset, allowing us to compare the performance of the proposed DNN classifier with our models for the detection of character-based DGA domain names. The DNN model performed slightly better in terms of ACC than the best model proposed in our work, which is the random forest classifier. However, it should be noted that the dataset used in our work to train and evaluate the models contains DGA domains used by 56 malware families.

The paper [11] proposed two DGA classifiers based on LSTM networks: binary classifiers, such as those presented in our work, and multi-class classifiers that allow a domain to be assigned to the specific DGA that generated it. The dataset included both word-based and character-based DGA domains.

3. Insights into the Machine Learning Models Used

This section provides a concise but informative overview of the machine learning algorithms used to detect DGA domains. Adapting our approach to the nuances of character-based and word-based DGA domains, we use classical methods such as logistic regression, Gaussian naive Bayes, support vector machine, random forest, extremely randomized trees, k-nearest neighbors, and nearest centroid for character-based DGA domains detection. In addition, long short-term memory (LSTM) and convolutional neural network (CNN) are specifically used for the complex task of word-based DGA detection.

3.1. Logistic Regression

Logistic regression is a type of regression algorithm tailored to be used in classification tasks. This algorithm can be used to compute the probability that an instance represented by the set of features belongs to a particular class. In the training process, we aim to optimize values of vector θ, which is a vector of weights plus the bias term, so that training instances that represent DGA domains are assigned class 1, and instances representing benign domains are assigned class 0. During the training, regularization terms like l1 or l2 can be employed, to prevent the model from overfitting [26].

3.2. Gaussian Naive Bayes

Gaussian naive Bayes is another example of a supervised machine learning algorithm that can be used for DGA detection and it is based on the Bayes theorem [27]. It is a model that predicts a class of instances based on conditional probabilities. During the training phase, the model builds probability distributions of values of features from training instances for two classes of domains (DGA and non-DGA). The likelihood of the features is assumed to be Gaussian [28]. After training, the model assigns a new instance x to either class calculating posterior probabilities that x belongs to DGA and non-DGA domains. The term "naive" comes from the fact that the model assumes feature-wise conditional independence given the class variable [28].

3.3. Support Vector Machine

Support vector machine (SVM) is an approach used for classification tasks and is frequently regarded as one of the best classifiers that do not require extensive customization [29]. SVM aims to construct a hyperplane of dimension $n-1$ given that, each instance in our dataset has n features in order to separate instances belonging to different classes. The classification decision is made by looking at which side of the hyperplane the instance lies.

The SVM classifier is sometimes called a soft margin classifier because the hyperplane it constructs has margins - that is perpendicular distance from some of the training observations [29]. The term "soft" comes from the fact the separating hyperplane may not perfectly separate two classes i.e., some training instances can be on the wrong side of the margin, but the obtained separation can generalize better on new instances. In some cases, the data points of two classes may even not be separable. With SVM, a kernel function can be applied to transform feature space. For instance, with a polynomial kernel, we can transform feature space into a higher dimension, where the separation of classes can be done more easily [26].

3.4. Random Forest

Random forest is an example of an ensemble learning algorithm that combines predictions of multiple decision trees.

Decision tree on the other hand is a very simple yet very powerful algorithm that can be used for classification tasks [26]. To train the decision tree in Scikit-learn the classification and regression tree (CART) algorithm is used.

The training process of the decision tree begins by splitting the training dataset into two subsets using a single feature f_n, and some threshold t_{f_n}. The algorithm selects the pair (f_n, t_{f_n}), that produces the lowest value of the cost function, which reflects the impurity of the resulting subsets. After the first split, the two nodes representing the two subsets of the training dataset are obtained. The splitting operation is then performed on these nodes and on the resulting nodes recursively.

The process stops if the split minimizing the cost function cannot be found or based on some criteria like the maximum depth of the tree or maximum number of leaves.

The random forest algorithm trains multiple instances of decision trees, introducing some randomness into the process of creating individual trees. The goal of this randomness is to obtain decision trees that produce as independent decisions as possible. One approach to introduce this randomness is to use different training datasets for each classifier or use random feature subset for splitting decisions. The predictions of each decision tree classifier are combined to give the final results. Such an approach often results in better accuracy than the best classifier in the ensemble [26].

3.5. Extremely Randomized Trees

Extremely randomized trees or extra trees are another example of the ensemble learning algorithm. This algorithm is basically a random forest algorithm that introduces more randomness into the process of building individual trees by using random thresholds for each feature rather, than searching for the threshold that best minimizes the cost function [26].

3.6. K-Nearest Neighbors

Classification using the k-nearest neighbors algorithm is a type of instance-based learning. This model stores the training instances of the training data and performs classification on new instances based on a majority vote of the k-nearest neighbors of training instances. An instance is assigned a class, that is a majority among k-nearest neighbors [28].

3.7. Nearest Centroid

The nearest centroid classifier is a very simple algorithm, which assigns each class a mean (centroid) of their instances. The class of the new instance is assigned based on the centroid of which class is closer to the new observation [28].

3.8. Convolutional Neural Networks

In the last few years convolutional neural networks (CNNs) have managed to achieve very good performance on some complex visual tasks, such as image classification [26]. Though CNNs originated from the exploration of the visual cortex of the brain, they can be deployed in other tasks, such as voice recognition or natural language processing (NLP) [26], [30].

CNNs use convolutional layers which consist of a set of filters also called kernels. The kernel can be treated as a matrix that is used in convolution operation with portions of the input sequence such as pixels of the image or an array of characters. The objective of this process is to extract patterns that are important for predictions. For example in the image processing task, the convolutional layer can extract some high-level features such as edges [31]. The values in the matrix representing the kernel, are learned during the training process.

3.9. Long Short-Term Memory Networks

Long short-term memory networks are a special type of recurrent neural networks (RNN) capable of learning long-term dependencies [32]. Traditional RNNs suffer from the problem of a vanishing and exploding gradient during backpropagation when dealing with more contextual data [32]–[34]. Long short-term memory networks, or simply LSTMs, use separate paths for long-term and short-term memory to avoid the vanishing and exploding gradient problem [33].

A single LSTM network module consists of three different gates that control the flow of information: forget gate, input gate, and output gate. LSTM networks are structured as chains of repeating modules. The output path from one module serves as the input for the corresponding path for the next module.

4. Proposed Models for Detecting Character-based DGA Domains

This section focuses on the use of classical machine learning methods for character-based DGA domains detection. We begin with a description of the dataset used. This is followed by a description of the process of constructing and selecting features to train the models. We conclude by describing the training process. The models were built using the Scikit-learn Python library.

4.1. Dataset

The dataset used consists of training and test subsets. Both contain non-DGA and DGA domains in a 1:1 ratio. A total of 450 000 benign domains (400 000 used for training and 50 000 for model evaluation) were derived from the top one million domain names ranked by Majestic [35]. As for the DGA domains, samples from 56 malware families were used. The DGA domains were obtained by executing reverse-engineered DGA code snippets available online or using predefined domain lists.

Table 2 shows the DGA datasets used to train and test the models. The source column serves as a reference as to where the domain names came from.

Tab. 2. Character-based and mixed DGA domains comprising the training and test dataset (C – character-based DGA, M – mixed DGA).

Malware family	Size of training dataset	Size of test dataset	Type	Source	Malware family	Size of training dataset	Size of test dataset	Type	Source
Orchard variant 3	9080	1135	C	[36]	Dyre	8860	1108	C	[39]
Vawtrak variant 1	9058	1132	C	[37]	Enviserv	8854	1107	C	[40]
Zeus Newgoz	9029	1129	C	[36]	Ranbyus variant 2	8850	1106	C	[37]
Qsnatch variant 1	9013	1127	C	[36]	Shiotob	8847	1106	C	[36]
Conficker	9007	1126	C	[38]	Chinad	8829	1104	C	[40]
Padcrypt v2.2.97.0	8993	1124	C	[36]	Cryptolocker	8825	1103	C	[38]
Ramdo	8989	1124	C	[37]	Murofet variant 3	8819	1102	C	[37]
Dircrypt	8989	1124	C	[36]	Vidro	8818	1102	C	[40]
Padcrypt v2.2.86.1	8987	1124	C	[36]	Pitou	8812	1101	C	[36]
Ramnit	8980	1122	C	[36]	Necurs	8772	1096	C	[36]
Qsnatch variant 2	8979	1122	C	[36]	Sisron	8544	1068	C	[36]
Tinba	8979	1122	C	[37]	Pykspa	8516	1065	C	[37]
Kraken variant 2	8978	1122	C	[36]	Banjori	6812	851	M	[36]
Fobber variant 2	8976	1122	C	[37]	Torpig	6157	770	C	[41]
Murofet variant 2	8969	1121	C	[37]	Mydoom	5592	699	C	[37]
Locky variant 3	8968	1121	C	[37]	Simda	5180	647	C	[36]
Kraken variant 1	8961	1120	C	[36]	Zloader	3139	392	C	[36]
Proslikefan	8954	1119	C	[36]	Tempedreve	2823	353	C	[37]
Locky variant 2	8931	1117	C	[37]	Sharkbot v2.8	2582	323	C	[36]
Symmi	8928	1116	C	[37]	Zeus	889	111	C	[38]
Pushdo	8925	1116	C	[37]	Sharkbot v1.63	348	44	C	[36]
Ranbyus variant 1	8922	1115	C	[37]	Sharkbot v0.0	317	40	C	[36]
Nymaim variant 1	8916	1115	C	[37]	Sharkbot v2.1	317	40	C	[36]
Qadars variant 3	8914	1114	C	[36]	Vawtrak variant 3	267	33	C	[36]
Verblecon	8896	1112	C	[37]	Vawtrak variant 2	267	33	C	[36]
Murofet variant 1	8881	1110	C	[37]	Ccleaner	149	19	C	[40]
Fobber variant 1	8876	1109	C	[37]	Alueron Dnschanger	4	1	C	[36]
Corebot	8867	1108	C	[36]	Total	400,000	50,000		
Qakbot	8866	1108	C	[37]					

4.2. Features

In the process of constructing the features, the approach chosen was to use the information contained in the domain name itself. Before extracting the features from the domain, each was stripped of its TLD, and the remaining labels were converted to lowercase and combined into a single string without dots. So, for example, the domain "gmail.google.com" would be converted to the string "gmailgoogle".

The approach of not including TLDs can be found in many works, but as the paper [21] shows, the domain TLD also contains information that can be useful for DGA detection. Besides, some TLDs are often associated with malicious activities [8]. Therefore, each TLD domain has been encoded with a single number and included as such in the feature set. Below is a list of the initial 35 proposed features. Features 1–22 and 29–30 were previously used in [4], [14], [19], while feature 35 was proposed in [21]. Feature 28 is the domain length and it was used in [8] and [42]. In addition, features 23–27, 31 and 33–34 were proposed in this work.

After applying feature elimination techniques, some of the features were removed from the final set. Although it may seem unnecessary to include them in the list below, since they were eventually removed, we retain their mention to provide readers with a complete description of the classifier development process.

Here is the list of features numbered as follows, along with a brief description:

1) count_2gram(d): A number of 2-grams of the domain name d, which are also found in the list of 500 most frequent 2-grams found in the 10 000 most popular non-DGA domains.
2) m_2gram(d): The 2-gram frequency distribution of the domain name d. $f(i)$ is the total number of occurrences of 2-gram found in the 500 most common 2-grams found in the 10 000 most popular non-DGA domain names. Index(i) is the rank of 2-gram among all total possible 2-grams found in the 10 000 most popular non-DGA domains. For example, if 2-gram "a0" is the second most popular 2-gram found in the 10 000 non-DGA domains it gets the rank of 2.
$$\sum_{i=1}^{\text{count\_2gram}(d)} f(i) \cdot \text{index}(i) \, .$$
3) s_2gram(d): The 2-gram weight of the domain name d. $vt(i)$ is the rank of 2-gram among the 500 most common 2-grams found in the 10 000 most popular non-DGA domain names.
$$\frac{\sum_{i=1}^{\text{count\_2gram}(d)} f(i) \cdot vt(i)}{\text{count\_2gram}(d)} \, .$$

4) ma_2gram(d): The average 2-gram frequency distribution of the domain name d. len_2gram(d) is the total number of 2-grams in d.

$$\frac{\text{m\_2gram}(d)}{\text{len\_2gram}(d)} .$$

5) sa_2gram(d): The average 2-gram weight distribution of the domain name d.

$$\frac{\text{s\_2gram}(d)}{\text{len\_2gram}(d)} .$$

6) tan_2gram(d): The average number of popular 2-grams in the domain name d.

$$\frac{\text{count\_2gram}(d)}{\text{len\_2gram}(d)} .$$

7) taf_2gram(d): The average frequency of popular 2-grams in the domain name d.

$$\frac{\sum_{i=1}^{\text{count\_2gram}(d)} f(i)}{\text{count\_2gram}(d)} .$$

8) count_3gram(d): A number of 3-grams of the domain name d, which are also found in the list of 500 most frequent 3-grams found in the 10 000 most popular non-DGA domains.

9) m_3gram(d): The 3-gram frequency distribution of the domain name d. $f(i)$ is the total number of occurrences of 3-gram found in the 500 most common 3-grams found in the 10 000 most popular non-DGA domain names. Index(i) is the rank of 3-gram among all total possible 3-grams found in the 10 000 most popular non-DGA domains. For example, if 3-gram "a0-" is the second most popular 3-gram found in 10 000 non-DGA domains it gets the rank of 2.

$$\sum_{i=1}^{\text{count\_3gram}(d)} f(i) \cdot \text{index}(i) .$$

10) s_3gram(d): The 3-gram weight of the domain name d. $vt(i)$ is the rank of 3-gram among the 500 most common 3-grams found in the 10 000 most popular non-DGA domain names.

$$\frac{\sum_{i=1}^{\text{count\_3gram}(d)} f(i) \cdot vt(i)}{\text{count\_3gram}(d)} .$$

11) ma_3gram(d): The average of 3-gram frequency distribution of the domain name d. len_3gram(d) is the total number of 3-grams in d.

$$\frac{\text{m\_3gram}(d)}{\text{len\_3gram}(d)} .$$

12) sa_3gram(d): The average of 3-gram weight distribution of the domain name d.

$$\frac{\text{s\_3gram}(d)}{\text{len\_3gram}(d)} .$$

13) tan_3gram(d): The average number of popular 3-grams in the domain name d.

$$\frac{\text{count\_3gram}(d)}{\text{len\_3gram}(d)} .$$

14) taf_3gram(d): The average frequency of popular 3-grams in the domain name d.

$$\frac{\sum_{i=1}^{\text{count\_3gram}(d)} f(i)}{\text{count\_3gram}(d)} .$$

15) tanv(d): The distribution of vowels in the domain name d. countv(d) is the number of vowels found in the domain d. len(d) is a total number of characters in d.

$$\frac{\text{countnv}(d)}{\text{len}(d)} .$$

16) tanco(d): The distribution of consonants in the domain name d. countco(d) is the number of consonants found in the domain d.

$$\frac{\text{countco}(d)}{\text{len}(d)} .$$

17) tandi(d): The distribution of digits in the domain name d. countdi(d) is the number of digits found in the domain d.

$$\frac{\text{countdi}(d)}{\text{len}(d)} .$$

18) tansc(d): The distribution of special characters in the domain name d. countsc(d) is the number of occurrences of the "-" character in d.

$$\frac{\text{countsc}(d)}{\text{len}(d)} .$$

19) tanhe(d): The distribution of hexadecimal characters in the domain name d. counthe(d) is the number of hexadecimal characters found in the domain d.

$$\frac{\text{counthe}(d)}{\text{len}(d)} .$$

20) ent_char(d): Character entropy of the domain name d. $D(x)$ is the probability distribution of the character x in the domain name d.

$$-\sum_{x} D(x) \cdot \log\big(D(x)\big) .$$

21) EOD(d): The expected value of the domain name d. $n(x)$ is the frequency of occurrence of character x in the domain name d, and $p(x)$ is the probability distribution of character x calculated based on the 10 000 most popular benign domains.

$$\frac{\sum_{x} n(x) \cdot p(x)}{\sum_{x} n(x)} .$$

22) is_first_char_digit(d): 1 if the first character of the domain d is a digit, else 0.

23) vcds_entropy(d): Only four categories of characters are considered in this entropy: digits, special character "–", consonants, and vowels. $K(k)$ is the probability distribution of category k in the domain name.

$$-\sum_{k}^{v,c,d,s} K(k) \cdot \log\big(K(k)\big) .$$

24) conditional_vcds_entropy(d): In this measurement, the domain name d is divided into 2-grams, which are denoted by a pair of categories k and l. The category can be a vowel, consonant, digit, or special character "–". $K(k|l)$ is the probability distribution of a 2-gram in which the first character belongs to category k and the second to category l. $K(k,l)$ is the probability distribution of a 2-gram in which one character belongs to the k category and the other to the l category.

$$-\sum_{k}^{v,c,d,s}\sum_{l}^{v,c,d,s} K(k,l)\cdot\log\frac{1}{K(k|l)}\,.$$

25) double_consonants(d): The number of occurrences of two consonants next to each other in the domain name d.

26) double_vowels(d): The number of occurrences of two vowels next to each other in the domain name d.

27) double_chars(d): The number of occurrences of two of the same characters next to each other in the domain name d.

28) len(d): The total number of characters in the domain name d.

29) entropy_2gram(d): The 2-gram entropy of the domain name d. $vt(i)$ is the rank of 2-gram among the 500 most common 2-grams found in the 10 000 most popular non-DGA domain names.

$$-\sum_{i=1}^{\text{count\_2gram}(d)}\frac{vt(i)}{500}\cdot\log\frac{vt(i)}{500}\,.$$

30) entropy_3gram(d): The 3-gram entropy of the domain name d. $vt(i)$ is the rank of 3-gram among the 500 most common 3-grams found in the 10 000 most popular non-DGA domain names.

$$-\sum_{i=1}^{\text{count\_3gram}(d)}\frac{vt(i)}{500}\cdot\log\frac{vt(i)}{500}\,.$$

31) vc_bigram_ratio(d): The ratio of the number of 2-grams that comprise a vowel-consonant or consonant-vowel pair $vc(d)$ to the number of 2-grams of the domain name d.

$$\frac{vc(d)}{\text{len\_2gram}(d)}\,.$$

32) tld: Encoded top level domain.

33) jsd_2gram(d): The 2-gram Jensen-Shannon divergence of the domain name d. $P(x)$ is the probability distribution of 2-grams in the domain name d. $Q(x)$ is the probability distribution of 2-grams in the 10 000 most popular domain names ranked by Majestic. $M(x)$ is a mixture distribution of P and Q.

$$\frac{1}{2}\sum_{x}P(x)\log\frac{P(x)}{M(x)}+\frac{1}{2}\sum_{x}Q(x)\log\frac{Q(x)}{M(x)}\,.$$

34) jsd_3gram(d): The 3-gram Jensen-Shannon divergence of the domain name d. $P(x)$ is the probability distribution of 2-grams in the domain name d. $Q(x)$ is the probability distribution of 3-grams in the 10 000 most popular domain names ranked by Majestic. $M(x)$ is a mixture distribution of P and Q.

$$\frac{1}{2}\sum_{x}P(x)\log\frac{P(x)}{M(x)}+\frac{1}{2}\sum_{x}Q(x)\log\frac{Q(x)}{M(x)}\,.$$

35) ji(d): Jaccard's index of the domain name d. 2grams(d) is the set of 2-grams that make up the domain name d. $DN_1, DN_2, \ldots, DN_{10000}$ are the 10 000 most popular domains ranked by Majestic.

$$\sum_{i=1}^{10000}\frac{|\text{2grams}(d)\cap\text{2grams}(DN_i)|}{|\text{2grams}(d)\cup\text{2grams}(DN_i)|}\,.$$

4.3. Feature Selection

In order to remove irrelevant and redundant features, a combination of feature selection techniques was employed to reduce the number of features from an initial set of 35. Firstly, ANOVA (analysis of variance) was utilized, which is particularly useful when dealing with categorical target variables and continuous features [43]. This statistical test helps identify features that are dependent on the target variable (class 0 or 1). Subsequently, the random forest and extra trees algorithms were leveraged to assess feature importance. By ranking features based on their contribution to predictive accuracy, this technique aids in the identification of less impactful features that may be considered for removal [26].

The last technique used for feature selection was to use the Pearson correlation method to calculate correlations between each pair of features. This helped to uncover and eliminate redundant features, that convey the same type of information.

As a starting point for eliminating redundant features, we took a correlation coefficient value greater than 0.9 or less than –0.9. Thus, we removed features 9, 11, 29, 30, and 33. Then, taking into account the results of the ANOVA test and the feature importance produced by random forest and extra trees algorithms, we got rid of more features through the process of elimination. In this way, features 7, 22, 25, 26, and 27, were removed. In this way, we removed 10 features from the initial set of 35.

4.4. Training

In the training phase of our character-based DGA detection models, we used a technique known as 10-fold cross-validation to select the best model hyperparameters. Dividing our dataset into ten groups, or folds, the learning process is repeated ten times, with each fold serving once as a test set. This strategy provides a more comprehensive assessment of model performance across different subsets of the data [26], [28], [29].

After selecting the best-performing hyperparameters, the models were re-trained using the entire training dataset.

Tab. 3. Word-based and mixed DGA domains comprising the training and test dataset (W – word-based DGA, M – mixed DGA).

Malware family	Size of training dataset	Size of test dataset	Type	Source
Nymaim variant 2	66853	8356	W	[36]
Banjori	17246	2156	M	[36]
Suppobox	66844	8356	W	[37]
Gozi	47939	5992	W	[37]
Matsnu	66912	8364	W	[38]
Rovnix	66770	8346	W	[38]
Bigviktor	66780	8348	W	[40]
Emotet	656	82	W	[44]
Total	400,000	50,000		

5. Proposed Models for Detecting Word-based DGA Domains

This section delves into the topic of word-based DGA detection, extending our exploration beyond character-based DGA classifiers. Using advanced neural network architectures, particularly long short-term memory networks and convolutional neural networks, the methodology for learning classifiers will be described. Initially, we describe the dataset used to train and evaluate the classifiers. We then examine the training process and delve into the architectural nuances of the models used. The models were constructed using the Keras library with the TensorFlow backend in Python.

5.1. Dataset

The dataset used to train and evaluate the classifiers includes a total of 900 000 domain names. Within this dataset, 450 000 benign domain names were taken from the one million most popular domain names ranked by Majestic [35], with a subset of 50 000 domains reserved for the purpose of model evaluation. At the same time, 450 000 domains associated with 8 different malware families were collected to form the DGA domain set. As in the dataset used to create the character-based DGA classifier, the DGA domains were obtained by executing reverse-engineered DGA code snippets available online or using predefined domain lists. Notably, similar to the approach taken in creating character-based DGA classifiers, one mixed-type DGA was included in the dataset. Details of the DGA domains used are shown in Tab. 3.

Listing 1: LSTM model code

```
model=Sequential(name='lstm_model')
model.add(Embedding(input_dim=56_000,
      output_dim=128, input_length=64))
model.add(LSTM(units=128, unroll=True,
      return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(units=128, unroll=True))
model.add(Dropout(0.2))
model.add(Dense(units=128, activation='relu',
      kernel_initializer='glorot_normal'))
model.add(Dropout(0.2))
model.add(Dense(units=64, activation='relu',
      kernel_initializer='glorot_normal'))
model.add(Dropout(0.2))
model.add(Dense(units=1, activation='sigmoid',
      kernel_initializer='glorot_normal'))
opt = Adam(learning_rate=0.005)
model.compile(loss='binary_crossentropy',
      optimizer=opt, metrics=['accuracy'])
```

5.2. Training

During the process of building the classifiers, different architectures were tested, and those that showed the best performance are presented in this paper. Code snippets showing the

Listing 2: CNN model code

```
model=Sequential(name='cnn_model')
model.add(Embedding(input_dim=56_000,
      output_dim=128, input_length=64))
model.add(Conv1D(200, 4, padding='same',
      activation='relu',
      kernel_initializer='glorot_normal'))
model.add(Dropout(0.5))
model.add(MaxPooling1D(pool_size=2, strides=2,
      data_format='channels_first'))
model.add(Conv1D(100, 2, padding='same',
      activation='relu'))
model.add(MaxPooling1D(pool_size=2, strides=2,
      data_format='channels_first'))
model.add(Dropout(0.5))
model.add(Flatten())
model.add(Dense(100, activation='relu',
      kernel_initializer='glorot_normal'))
model.add(Dropout(0.5))
model.add(Dense(10, activation='relu',
      kernel_initializer='glorot_normal'))
model.add(Dense(1, activation='sigmoid',
      kernel_initializer='glorot_normal'))
model.compile(loss='binary_crossentropy',
      optimizer='adam', metrics=['accuracy'])
```

Tab. 4. Compilation of evaluation metrics for our proposed models and state-of-the-art solutions (C – character-based DGA, W – word-based DGA, G – general approach for detecting both word-based and character-based DGA domain names).

Classifier	Type	PPV	TPR	FPR	FNR	F1	ACC	AUC
RF [19]	C	90.7%	91%	9.3%		90.8%	90.8%	
RF [14]	G	97.08%	96.98%	2.92%	3.02%	97.03%	97.03%	
N-Gram [21]	G			6.14%	7.42%		94.04%	
LSTM [22]	C	95.05%	95.14%			94.58%	95.14%	
RF [23]	W						78.2%	
J48 [9]	W	98.25%	95.81%	1.78%	4.19%	97.01%	96.99%	
MLP [15]	G	99.5%	99.55%			99.5%	99.5%	
LSTM [18]	G	98.43%	98.4%			98.42%		
ATT-CNN-BiLSTM [24]	G	99.01%	99.07%			98.79%	98.82%	0.9990
CNN+LSTM [17]	W	95.57%	97.66%	4.54%		96.6%	96.56%	0.9944
DNN [25]	C	89.24%	99.14%				97.79%	0.9900
LSTM [11]	G	96.74%	85.71%			89.13%		0.9993
The proposed models in our work								
ET	C	96.84%	95.48%	3.12%	4.52%	96.16%	96.18%	0.9937
SVM	C	94.39%	93.08%	5.53%	6.92%	93.73%	93.77%	0.9846
LR	C	94.28%	93.18%	5.66%	6.82%	93.72%	93.76%	0.9847
GNB	C	83.28%	91.27%	18.33%	8.73%	87.09%	86.47%	0.9278
NC	C	85.36%	86.50%	14.83%	13.50%	85.93%	85.83%	0.9380
RF	C	97.60%	96.43%	2.37%	3.57%	97.01%	97.03%	0.9954
KNN	C	96.33%	94.98%	3.62%	5.02%	95.65%	96.00%	0.9901
LSTM	W	94.34%	96.50%	5.78%	3.50%	95.41%	95.36%	0.9905
CNN	W	97.84%	98.78%	2.19%	1.22%	98.31%	98.30%	0.9975

definitions of LSTM and CNN models in Python using the Keras library are shown in Listing 1 and 2, respectively.

The process of training the neural networks began with proper domain preparation, which involved converting them to lowercase, removing TLDs, and then breaking them down into lists of words that make up the domain name using the wordninja package in Python [45]. For example, the domain "watchfire.com" would be converted to a list ["watch", "fire"]. Words to be input to neural networks must be uniquely encoded. During data processing, it was determined that 56 000 unique values could be used to encode all the words making up the training and test set. In fact, in the dataset, the number of unique words forming all domains was 55 027, but experiments showed that the size of the vocabulary does not affect the performance of the models, so for simplicity it was decided to set the number known in the Keras embedding layer as input_dim, which refers to the size of the vocabulary, to 56 000. Similarly, we set the maximum number of words a domain can contain to 64 (input_length parameter in the embedding layer).

Both CNN and LSTM models have an embedding layer that learns to map the corresponding values representing the words that make up the domain to 128-dimensional vectors. A number of works, such as [11], [13], [16]–[18], [46], use deep learning models with embedding layers that rely on character embedding, which is different from the approach we used.

The LSTM model implements two long short-term memory layers, each consisting of 128 units. The first LSTM layer is set to return the full sequence of output data for each time step. By applying dropout layers at a rate of 20% after each LSTM layer, the model alleviates over-fitting during learning.

Following the LSTM layers are three dense layers, containing 128, 64, and 1 unit(s), respectively. The first two use the rectified linear unit (ReLU) activation function. These dense layers are interspersed with dropout layers to increase the robustness of the model. The last dense layer with a sigmoidal activation function makes the model act as a binary classifier.

The CNN model employs two Conv1D layers containing 200 and 100 filters, respectively. These convolution layers are activated using the ReLU function. Dropout layers with a 50% dropout rate follow each convolution layer to mitigate over-fitting. The convolution layers are followed by two Max-

Tab. 5. Accuracy for each character-based DGA detection model relating to different DGA families and benign domains.

Origin of domains	ET [%]	SVM [%]	LR [%]	GNB [%]	NC [%]	RF [%]	KNN [%]
Majestic	96.88	94.47	94.34	81.67	85.17	97.63	96.38
Orchard v3	99.47	97.00	97.27	100.00	99.56	99.65	99.65
Vawtrak v1	92.49	92.49	92.84	89.93	85.34	93.82	93.29
Zeus Newgoz	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Qsnatch v1	91.30	92.64	92.81	94.94	93.70	92.19	89.97
Conficker	82.15	82.33	82.24	92.63	88.10	84.81	79.31
Padcrypt v2.2.97.0	98.67	98.49	98.58	92.35	75.09	99.11	98.93
Ramdo	99.38	99.82	99.73	93.24	78.11	99.47	99.29
Dircrypt	97.86	97.78	97.69	96.26	92.70	98.04	96.89
Padcrypt v2.2.86.1	98.84	98.58	98.58	92.35	76.25	99.64	99.64
Kraken v2	94.56	92.16	92.25	94.39	91.00	95.37	92.78
Ramnit	99.55	98.66	98.75	98.04	94.39	99.82	98.66
Qsnatch v2	50.53	40.29	39.75	79.14	80.66	60.96	48.31
Fobber v2	96.17	95.01	95.37	94.74	90.29	96.79	94.56
Tinba	97.86	97.95	98.13	96.88	91.62	97.42	96.52
Murofet v2	99.91	99.82	99.73	98.39	95.90	99.82	99.38
Locky v3	96.79	95.36	95.81	95.63	92.15	96.97	95.09
Kraken v1	96.96	97.14	97.14	93.66	86.61	97.14	97.05
Proslikefan	89.54	87.31	87.40	92.14	89.28	91.42	86.68
Locky v2	88.90	88.81	88.81	93.82	90.06	90.87	87.38
Symmi	95.97	97.49	97.49	75.36	49.28	98.12	97.31
Pushdo	83.96	71.68	72.67	55.20	42.47	98.57	90.50
Ranbyus v1	99.10	99.64	99.64	97.22	92.20	99.01	98.92
Nymaim v1	88.25	86.01	85.83	92.65	87.98	88.07	84.30
Qadars v3	99.64	99.73	99.73	99.46	95.78	99.64	99.37
Verblecon	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Murofet v1	99.91	100.00	100.00	99.91	99.55	99.91	99.91
Fobber v1	99.91	100.00	100.00	99.01	96.30	100.00	99.91
Dyre	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Qakbot	99.10	98.74	98.74	97.74	94.95	99.19	98.65
Corebot	100.00	100.00	100.00	99.91	100.00	100.00	100.00
Enviserv	99.82	99.91	99.91	100.00	99.55	99.91	99.73
Shiotob	99.19	99.82	99.82	99.64	98.01	99.19	98.19
Ranbyus v2	99.55	100.00	100.00	97.92	93.58	99.64	99.19
Chinad	100.00	100.00	100.00	99.91	99.73	99.91	99.73
Cryptolocker	99.09	99.37	99.27	96.46	91.30	99.09	98.01
Murofet v3	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Vidro	95.55	94.28	94.37	96.64	92.83	94.74	93.92
Pitou	96.91	61.22	63.03	46.23	30.52	98.27	98.27
Cecurs	97.35	96.44	96.53	95.62	91.24	97.35	96.81
Sisron	100.00	100.00	100.00	100.00	98.60	100.00	100.00
Pykspa	89.58	83.38	83.47	75.68	69.95	91.17	86.67
Banjori	100.00	94.24	93.07	0.00	0.00	99.88	100.00
Torpig	98.44	93.77	94.29	93.12	85.19	98.83	97.01
Mydoom	93.13	88.27	88.84	77.68	67.24	94.85	93.85
Simda	94.44	55.64	55.49	93.82	92.43	93.97	94.44
Zloader	99.74	100.00	100.00	98.72	96.17	99.74	99.49
Tempedreve	90.08	87.82	88.95	90.08	86.69	90.93	89.24
Sharkbot v2.8	100.00	100.00	100.00	100.00	100.00	100.00	99.69
Zeus	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Sharkbot v1.63	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Sharkbot v2.1	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Sharkbot v0.0	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Vawtrak v3	18.18	21.21	24.24	36.36	30.30	15.15	3.03
Vawtrak v2	36.36	30.30	36.36	48.48	33.33	36.36	30.30
Ccleaner	100.00	100.00	100.00	100.00	100.00	100.00	100.00
Alueron Dnschanger	100.00	100.00	100.00	100.00	100.00	100.00	100.00

Pooling1D layers with pool sizes of 2 and steps of 2. The CNN model, like the LSTM, adds 3 dense layers. The first two contain 100 and 10 units respectively with ReLU activation functions and one dropout layer. The last dense layer with a sigmoidal activation function transforms the model into a binary classifier. Dense layers in both models and convolution layers in the CNN model are initialized with glorot normal weights.

The compilation of both models uses binary cross-entropy loss, Adam's optimizer, and accuracy as evaluation factors. The LSTM model set the learning rate to 0.005, while the CNN model used the default value of 0.001. The LSTM model was trained for 5 epochs with a batch size of 128, while the CNN model underwent a longer learning period of 10 epochs under the same batch size conditions.

6. Evaluation of the Models

After the training process, we proceeded to evaluate models' performance using a dedicated test dataset containing 100 000 domain names. To measure the performance of the models, we used seven key metrics: ACC (overall accuracy), PPV (positive predictive value) or precision, TPR (true positive rate) or recall, FPR (false positive rate), FNR (false negative rate), F1 score and AUC (area under the ROC curve). The formulas used to calculate these indicators are shown below:

$$PPV = \frac{TP}{TP + FP} \cdot 100\% \,, \tag{1}$$

$$TPR = \frac{TP}{TP + FN} \cdot 100\% \,, \tag{2}$$

$$FPR = \frac{FP}{TN + FP} \cdot 100\% \,, \tag{3}$$

$$FNR = \frac{FN}{TP + FN} \cdot 100\% \,, \tag{4}$$

$$F1 = \frac{2TP}{2TP + FP + FN} \cdot 100\% \,, \tag{5}$$

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \cdot 100\% \,, \tag{6}$$

$$AUC = \int_0^1 TPR(FPR)\, dFPR \,. \tag{7}$$

Where true positives (TP) is the number of DGA domains that were classified correctly, true negatives (TN) is the number of benign domains that were classified correctly, false positives (FP) is the number of benign domains that were misclassified as DGA, and false negatives (FN) is the number of DGA domains that were classified as benign domains.

The specific values of these performance indicators for each model for detecting both character-based and word-based DGA are summarized at the bottom of the Tab. 4.

Tab. 6. Accuracy for each word-based DGA detection model relating to different DGA families and benign domains.

Origin of domains	LSTM [%]	CNN [%]
Majestic	94.22	97.81
Matsnu	99.76	99.88
Nymaim v2	89.64	96.46
Suppobox	97.69	99.77
Bigviktor	98.22	99.82
Rovnix	99.84	99.96
Gozi	93.56	96.81
Banjori	100.00	100.00
Emotet	9.76	17.07

7. Conclusions

In the case of detecting domains generated by character-based DGAs, five of the seven classifiers proposed in this work achieved ACC greater than 90%. The random forest-based model was the best classifier, achieving an ACC of 97.03% with fairly low FPR and FNR. Random forest has already been successfully used in other works, such as [14], [19] and [23]. The worst performing classifiers were those based on the gaussian naive Bayes model and the nearest centroid model, achieving ACCs of 86.47% and 85.83%, respectively. Table 5 also shows that these two models were unable to detect any of the 851 domains belonging to the mixed DGA used by the *Banjori* malware.

All models performed equally poorly in detecting DGA domains belonging to *Vawtrak* v2 and *Vawtrak* v3 malware, achieving an accuracy of less than 50%. If we look at the domains generated by these two variants, we can see that they are quite pronounceable character-based DGA domains. Examples of domains belonging to *Vawtrak* v2 include "alohgufda.com", "usornatda.com", or "fosornom.com", examples of *Vawtrak* v3 domains include "sumiwgecoll.com", "aldemegnehi.com", and "garidsemogn.com".

As for the task of detecting word-based DGA domains, both proposed classifiers achieved ACC scores greater than 95%. The CNN layer-based model was the best, achieving an ACC of 98.30% with low FPR and FNR values of 2.19% and 1.22% respectively. Table 6 shows that the model additionally achieved accuracy close to 100% for all DGAs except one belonging to the *Emotet* malware. The test dataset included 82 domains belonging to this DGA, and both the LSTM and CNN models performed poorly in detecting them, failing to exceed an accuracy of 20%.

This result may have been influenced by the fact that many of the domains generated by the Emotet DGA, such as "www.69po.com", "ceylonsri.com", or "senteum.com", are quite short in contrast to other DGAs and more closely resemble domains generated by character-based DGAs.

References

[1] "Botnet Threat Update Q3 2023", Spamhaus, [Online]. Available: https://info.spamhaus.com/botnet-threat-updates.

[2] "What is a Botnet?", Palo Alto Networks, [Online]. Available: https://www.paloaltonetworks.com/cyberpedia/what-is-botnet.

[3] A. Randall *et al.*, "The Challenges of Blockchain-based Naming Systems for Malware Defenders", *2022 APWG Symposium on Electronic Crime Research (eCrime)*, Boston, USA, 2022 (https://doi.org/10.1109/eCrime57793.2022.10142131).

[4] X.H. Vu, X.D. Hoang, and T.H.H. Chu, "A Novel Model Based on Ensemble Learning for Detecting DGA Botnets", *2022 14th International Conference on Knowledge and Systems Engineering (KSE)*, Nha Trang, Vietnam, 2022 (https://doi.org/10.1109/KSE56063.2022.9953792).

[5] E. Durmaz, "DGA Classification and Detection for Automated Malware Analysis", Cyber.WTF, 2017 [Online]. Available: https://cyber.wtf/2017/08/30/dga-classification-and-detection-for-automated-malware-analysis.

[6] "Kaspersky Security Bulletin 2023", Kaspersky, [Online]. Available: https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2023/11/28102415/KSB_statistics_2023_en.pdf.

[7] L. Asher-Dotan, "What is Domain Generation Algorithm: 8 Real World DGA Variants", Cybereason, [Online]. Available: https://www.cybereason.com/blog/what-are-domain-generation-algorithms-dga.

[8] R. Sivaguru *et al.*, "Inline Detection of DGA Domains Using Side Information", *IEEE Access*, vol. 8, pp. 141910–141922, 2020 (https://doi.org/10.1109/access.2020.3013494).

[9] X.D. Hoang and X.H. Vu, "A Novel Machine Learning-based Approach for Detecting Word-based DGA Botnets", *Journal of Theoretical and Applied Information Technology*, vol. 99, no. 24, 2021.

[10] D. Plohmann *et al.*, "A comprehensive measurement study of domain generating malware", *Proc. of 25th USENIX Security Symposium*, Austin, USA, pp. 263–278, 2016.

[11] J. Woodbridge, H.S. Anderson, A. Ahuja, and D. Grant, "Predicting Domain Generation Algorithms with Long Short-Term Memory Networks", *arXiv,* 2016 (https://doi.org/10.48550/arXiv.1611.00791).

[12] M. Pereira *et al.*, "Dictionary Extraction and Detection of Algorithmically Generated Domain Names in Passive DNS Traffic", *International Symposium on Research in Attacks, Intrusions, and Defenses*, Heraklion, Greece, 2018 (https://doi.org/10.1007/978-3-030-00470-5_14).

[13] R.R. Curtin *et al.*, "Detecting DGA Domains with Recurrent Neural Networks and Side Information", *Proc. of the 14th International Conference on Availability, Reliability and Security – ARES'19*, pp. 1–10, 2019 (https://doi.org/10.1145/3339252.3339258).

[14] X.D. Hoang and X.H. Vu, "An Improved Model for Detecting DGA Botnets Using Random Forest Algorithm", *Information Security Journal: A Global Perspective*, vol. 31, no. 4, pp. 441–450, 2021 (https://doi.org/10.1080/19393555.2021.1934198).

[15] A. Cucchiarelli, C. Morbidoni, L. Spalazzi, and M. Baldi, "Algorithmically Generated Malicious Domain Names Detection Based on n-Grams Features", *Expert Systems with Applications*, vol. 170, art. no. 114551, 2021 (https://doi.org/10.1016/j.eswa.2020.114551).

[16] B. Yu *et al.*, "Inline DGA Detection with Deep Networks", *2017 IEEE International Conference on Data Mining Workshops (ICDMW)*, New Orleans, USA, 2017 (https://doi.org/10.1109/ICDMW.2017.96).

[17] K. Highnam, D. Puzio, S. Luo, and N.R. Jennings, "Real-time Detection of Dictionary DGA Network Traffic Using Deep Learning", *SN Computer Science*, vol. 2, art. no. 110, 2021 (https://doi.org/10.1007/s42979-021-00507-w).

[18] D. Tran *et al.*, "A LSTM Based Framework for Handling Multiclass Imbalance in DGA Botnet Detection", *Neurocomputing*, vol. 275, pp. 2401–2413, 2018 (https://doi.org/10.1016/j.neucom.2017.11.018).

[19] X.D. Hoang and Q.C. Nguyen, "Botnet Detection Based on Machine Learning Techniques Using DNS Query Data", *Future Internet*, vol. 10, art. no. 43, 2018 (https://doi.org/10.3390/fi10050043).

[20] S. Yadav, A.K.K. Reddy, A.L.N. Reddy, and S. Ranjan, "Detecting Algorithmically Generated Malicious Domain Names", *Proc. of the 10th ACM SIGCOMM Conference on Internet Measurement*, pp. 48–61, 2010 (https://doi.org/10.1145/1879141.1879148).

[21] H. Zhao, Z. Chang, G. Bao, and X. Zeng, "Malicious Domain Names Detection Algorithm Based on N-Gram", *Journal of Computer Networks and Communications*, pp. 1–9, 2019 (https://doi.org/10.1155/2019/4612474).

[22] Y. Qiao *et al.*, "DGA Domain Name Classification Method Based on Long Short-term Memory with Attention Mechanism", *Applied Sciences*, vol. 9, no. 20, art. no. 4205, 2019 (https://doi.org/10.3390/app9204205).

[23] L. Yang *et al.*, "A Novel Detection Method for Word-based DGA", *Lecture Notes in Computer Science*, vol. 11064, pp. 472–483, 2018 (https://doi.org/10.1007/978-3-030-00009-7_43).

[24] F. Ren, Z. Jiang, X. Wang, and J. Liu, "A DGA Domain Names Detection Modeling Method Based on Integrating an Attention Mechanism and Deep Neural Network", *Cybersecurity*, vol. 3, art. no. 4, 2020 (https://doi.org/10.1186/s42400-020-00046-6).

[25] Y. Li, K. Xiong, T. Chin, and C. Hu, "A Machine Learning Framework for Domain Generation Algorithm (DGA)-based Malware Detection", *IEEE Access*, vol. 7, pp. 32765–32782, 2019 (https://doi.org/10.1109/access.2019.2891588).

[26] A. Géron, *Hands-on Machine Learning with Scikit-learn, Keras, and TensorFlow*, O'Reilly Media, Inc, 2nd ed., 848 p., 2019 (ISBN: 9781492032649).

[27] A. Smola and S.V.N. Vishwanathan, *Introduction to Machine Learning*, Cambridge University Press: Cambridge, UK, 2008.

[28] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python, *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011 (https://dl.acm.org/doi/10.5555/1953048.2078195).

[29] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning with Applications in R*, Springer, New York, 440 p., 2013 (https://doi.org/10.1007/978-1-4614-7138-7).

[30] K. Fukushima, "Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position", *Biological Cybernetics*, vol. 36, pp. 193–202, 1980 (https://doi.org/10.1007/BF00344251).

[31] S. Saha, "A Comprehensive Guide to Convolutional Neural Networks the ELI5 way", Saturn Cloud, 2018 [Online], Available: https://saturncloud.io/blog/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way.

[32] C. Olah, "Understanding LSTM Networks", Colah, 2015 [Online], Available: https://colah.github.io/posts/2015-08-Understanding-LSTMs/.

[33] J. Starmer, "Long Short-Term Memory (LSTM), Clearly Explained", StatQuest with Josh Starmer, 2022 [Online], Available: https://www.youtube.com/watch?v=YCzL96nL7j0.

[34] S. Hochreiter and J. Schimdhuber, "Long Short-Term Memory", *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997 (https://doi.org/10.1162/neco.1997.9.8.1735).

[35] "The Majestic Million", Majestic, [Online], Available: https://majestic.com/reports/majestic-million.

[36] J. Bader, "Binary Reverse Engineering Blog", [Online], Available: https://bin.re/blog.

[37] J. Bader, "Domain Generation Algorithms", GitHub repository, [Online], Available: https://github.com/baderj/domain_generation_algorithms.

[38] A. Abakumov, "DGA", GitHub repository, [Online], Available: https://github.com/andrewaeva/DGA.

[39] F. Denis, "Dyre/Dyreza DGA", GitHub repository, [Online], Available: https://gist.github.com/jedisct1/33ab6b4e81209dbf53a3.

[40] "DGA", GitHub repository, [Online], Available: https://github.com/360netlab/DGA.

[41] P. Chaignon, "DGA-collection", GitHub repository, [Online], Available: https://github.com/pchaigno/dga-collection.

[42] T.D. Truong and G. Cheng, "Detecting Domain-flux Botnet Based on DNS Traffic Features in Managed Network", *Security and Communi-*

cation Networks, vol. 9, pp. 2338–2347, 2016 (https://doi.org/10.1002/sec.1495).

[43] J. Brownlee, "How to Perform Feature Selection with Numerical Input Data", Machine Learning Mastery, [Online], Available: https://machinelearningmastery.com/feature-selection-with-numerical-input-data/.

[44] D. Takahashi, "Emotet Domain", GitHub repository, [Online], Available: https://github.com/HASH1da1/emotet-domain.

[45] Wordninja 2.0.0., Python Package Index, [Online], Available: https://pypi.org/project/wordninja/.

[46] R. Sivaguru *et al.,* "An Evaluation of DGA Classifiers", *2018 IEEE International Conference on Big Data (Big Data)*, Seattle, USA, 2018 (https://doi.org/10.1109/BigData.2018.8621875).

Hubert Biros
Independent Researcher, Kraków, Poland
E-mail: hubertbiros00@gmail.com

Mirosław Kantor, Ph.D.
Institute of Telecommunications
https://orcid.org/0000-0002-3160-6422
E-mail: miroslaw.kantor@agh.edu.pl
AGH University of Krakow, Kraków, Poland
https://www.agh.edu.pl

Staying Hidden at Battlefields While Communicating via Unmanned Vehicles

Karol Zientarski, Mykyta Muravytskyi, Krzysztof Skos, Kamil Chełminiak, and Pawel Kulakowski

AGH University of Krakow, Kraków, Poland

https://doi.org/10.26636/jtit.2025.FITCE2024.2086

Abstract — History shows that information is one of the key factors in military conflicts. During military conflicts, there is a need to maintain a communication channel on the battlefield while staying hidden from the enemy. In this paper, we present a simulator that allows to use a communication network and minimize the risk of being detected by the enemy. The simulator, using the Prim algorithm and fine-tuning, shows how a mobile ad-hoc network established between soldiers with the aid of unmanned vehicles, i.e. drones, may become undetectable for the enemy by properly optimizing drone positions.

Keywords — low probability detection, MANET, Prim algorithm, RSSI, UxV

1. Introduction

Mobile ad-hoc networks (MANETs) are commonly used in military scenarios, as they provide the flexibility required to accommodate a dense, chaotic and heterogeneous topology and are capable of operating in areas without any infrastructure.

In this paper, we take a closer look at a tactical MANET network composed of military units connected through a radio channel. The ad-hoc approach brings lots of complications including, but not limited to, complex routing, neighbor detection, and mobility issues. However, our work focuses on providing a disguise for communication, thus lowering the probability of detection (LPD) of the network [1].

Many factors affect the ability of an adversary to detect a radio transmission. Regardless of these, reducing the power received by the foe will make the detection task more difficult. This could even result in bringing the received power below the detection threshold of the adversary's receiver, thus making it impossible to detect the transmission. Although reducing transmission power limits the probability of it being detected, the network must remain operational, so that all units can still communicate with one another.

Despite that, the performance of almost any ad-hoc network can be enhanced using unmanned vehicles [2]–[4] (UxV, where "x" stands for one of the four types of vehicles – air [5], ground, surface, or underwater), especially in warfare conditions where their advantages are undeniable. Our goal is to design an algorithm that deploys UxVs in such a way that connectivity within the network is increased [6] and, more crucially, it allows the transmission from a potential adversary.

This paper is based on the presentation made at the 63rd FITCE 2024 international congress and titled "Hiding Radio Communication at Battlefields Using Unmanned Vehicles". However, the scope of our work is more extensive, as it includes algorithm details, additional scenarios, and comprehensive analysis of results, along with an in-depth discussion. The rest of this paper is organized as follows. In Section 2, we discuss previous works related to our topic. In Section 3, we explain the scenario and methodology of our investigation. In Section 4, we describe the LPD optimization algorithm. In Sections 5 and 6, we discuss the results of the research and their impact on the topic, respectively.

2. Related Work

There are several approaches to reducing the probability of detection of a network (LPD). For example, [7] presents an LPD algorithm for mobile networks, including field tests, based on the received signal strength indicator (RSSI) with a combination of the minimum spanning tree (MST) topology, referred to as RSSI distributed MST (RDMST). In [8], the authors explore possibilities of minimizing area coverage with enemy unit avoidance. Furthermore, in [9], a novel device capable of emulating networks for LPD problems is shown.

Furthermore, in [10], an idea of an LPD mobile network using UAV swarms was proposed based on numerous aerial devices that create, from scratch, and entire network that is hidden from enemy ground units. However, it is assumed that the distance from adversaries is known and calculated from RSSI measurement at the enemy receivers.

The topic of MANET networks covers a broad range of issues, and some authors conducted valuable research in the form of surveys. In [11], the authors discussed recent advances in protocol development and MANET applications. Next, it is known that the development of machine learning and artificial intelligence creates new possibilities for network optimization. In [12], AI-based MANET routing protocols, including both machine learning and biologically inspired approaches, are discussed.

On the other hand, the review featured in [13] presents a different approach to MANET cybersecurity, discussing a galore

This work is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) License.
For more information, see https://creativecommons.org/licenses/by/4.0/

Tab. 1. Simulation parameters.

Parameter	Value
Infantry Tx power	25 mW
Infantry radio range	25 km
Vehicle Tx power	63 mW
Vehicle radio range	40 km
UxV Tx power	143 mW
UxV radio range	60 km
Number of ally units	10
Number of ally infantry units	8
Number of ally vehicle units	2
Number of enemy units	3
Number of UxV adding cycles	6
Gradient descent iterations	1000
Map size	100 ×100 km
Map fraction occupied by allies	90%
Map fraction occupied by enemies	30%
Precision for map coverage calculations	0.2 km
Gradient learning rate	6

of security issues and cyberattacks aimed at MANET susceptibilities. The problem of hiding a whole network is only one of the vulnerabilities addressed in this paper. However, it is important to consider other cybersecurity challenges when designing a MANET.

In this work, we use the Prim algorithm to create minimum spanning trees (MSTs) [14]. Despite being an old approach, it is still used for present applications [15]. It is known that there are numerous other algorithms suitable for MST creation, e.g. Kruskal or Boruvka [16]. However, when comparing time complexities, Prim's is:

$$\mathrm{O}\big((V-1)\log(V)+E\log(V)\big)$$

and Kruskal's is:

$$\mathrm{O}\big(E\log(E)+V\log(V)\big)\,,$$

where E is the number of edges and V is the number of vertices. Furthermore, in practice, the complexities can be simplified to $\mathrm{O}(E\log(V))$ and $\mathrm{O}(E\log(E))$, respectively [17].

When the initial number of edges is much greater than the number of initial vertices, the Prim algorithm is more efficient [18]. Boruvka's algorithm has a time complexity of $\mathrm{O}(E\log(V))$, making it only as fast as Prim's algorithm [19].

3. Research Methodology and Scenario

We created a Python simulator that generates the required ally and enemy units on the battlefield. The goal of our study was to create a network between allies that was hard to detect by their enemies. The simulation parameters are presented in Tab. 1.

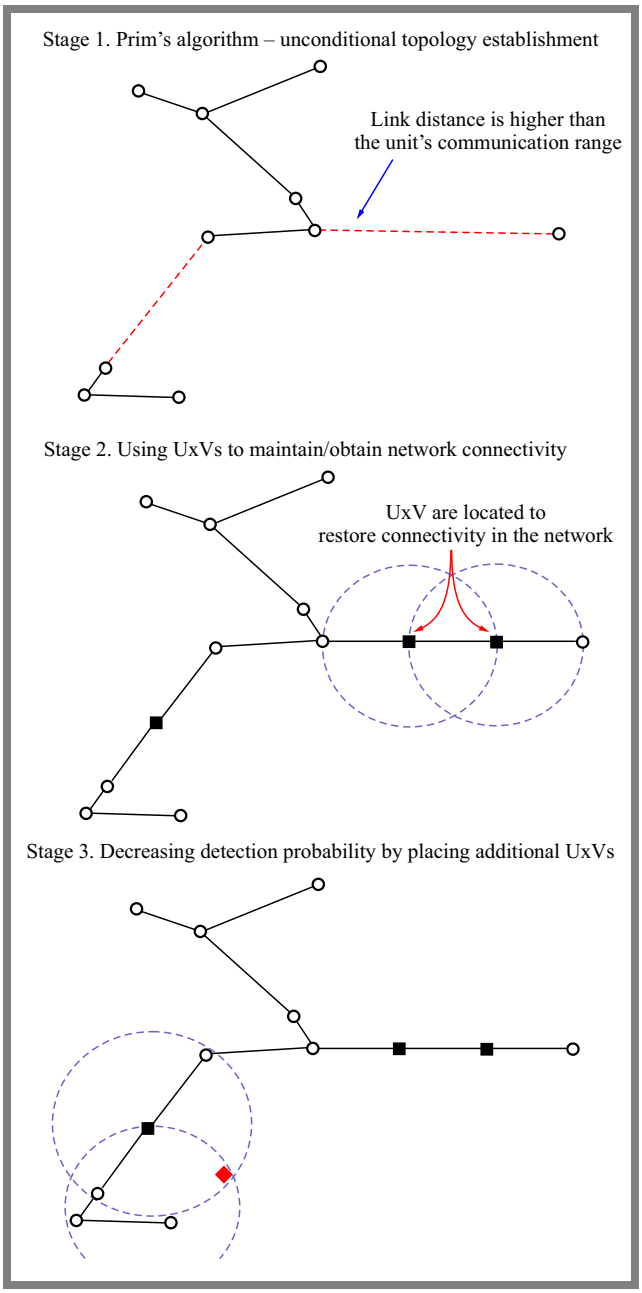


Fig. 1. Three stages of the network optimization process: 1) generating allied units generating allied units and building the spanning tree, 2) obtaining connectivity, and 3) optimizing the network. Legend: a black circle means an allied unit, a black square is UxV, a red rectangle stands for an enemy unit, a black line illustrates a connection between allies, a red dotted line identifies a connection between the allies that is longer than the allies' range.

Scenario assumptions:

1) The exact position of all the units (allied and enemy) is known, e.g. from GPS, satellite images, or other military tracking technologies.
2) We use the Friis loss model and calculate the power at the receiver of each unit.

3) We select a threshold detection power value according to an exemplary radio communicator used in military communication, equaling –110 dBm.
4) Radio units are portable and have a finite battery life, hence the requirement to limit the maximum transmission power.

In addition, we distinguish four types of units. Three types of allied units (i.e. the infantry, vehicles, and UxVs) and enemy units. All of them have radio stations with a receiver sensitivity of –110 dBm, operating at the 1.5 GHz frequency. The transmitter (Tx) powers were adapted to match the desired range in the medium with the Friis loss model. The detailed Tx specification is shown in Tab. 1.

4. Algorithm

4.1. Generating Units

Allied and enemy units are generated on a square field using log-normal distribution. Allies are placed on the left 90% of the field, and enemy units are placed on the right. Therefore, there exists a 20% of the area where all units have a chance of being positioned.

For example, for a field that is 100 km long, the allies may be generated within 0 to 90 km, and the enemies might be generated from 70 to 100 km. Furthermore, the ratio between allied vehicles and infantry units is 1:4.

4.2. Building the Spanning Tree Between the Stations

After unit generation, the Prim algorithm [14] is used to build the spanning tree for existing ally nodes and establish connectivity throughout the entire network. It is a greedy algorithm that finds a minimum spanning tree for a weighted undirected graph. Given a matrix of points, the algorithm starts with a designated point and an empty list of visited nodes.

In the subsequent steps, starting from the designated point, the algorithm picks an edge with the smallest weight connected to an unvisited node. Then, the newly connected node becomes designated. The algorithm ends when all nodes are visited and a connected graph with no cycles is created.

As an edge weight in the Prim algorithm, we use the respective distance between two nodes. Thus, in general, we decrease the probability of choosing longer edges and minimize the power levels of Tx. The topology of the network is shown in Fig. 1, stage 1.

In the next step, we add UxVs to the radio links, where there is no connectivity between units. We distinguish three distinct cases of UxV deployment:

1) If the sum of the unit's radio ranges is larger than the distance between them, we add only one UxV in the middle, in between the stations.
2) If the sum of the unit's radio ranges is smaller than the distance between them, we add two UxVs at the ends of the unit's radio ranges.

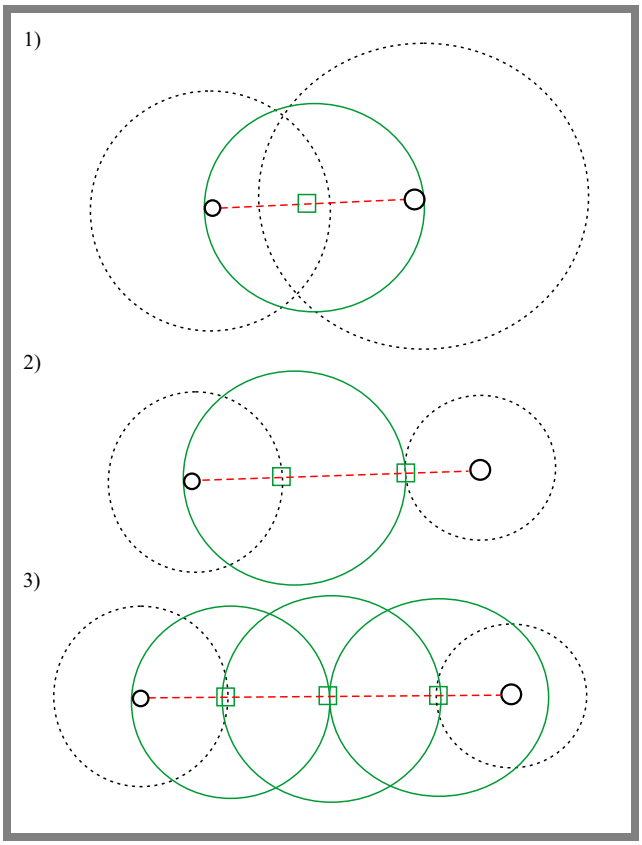


Fig. 2. Three possible scenarios taken into consideration during the network design phase. Legend: a black circle shows an allied unit, a dashed circle stands for allied radio range, a green square is an UxV, a green circle covers the UxV radio range, a red dotted line identifies the connection between allies that is longer than the range of the ally.

3) In case when the sum of the unit's radio ranges is smaller than the distance between them and the sum of the two UxV's radio ranges is smaller than the distance between units decreased by the sum of the unit's radio ranges, two UxVs are added on the ends of the unit's radio ranges, and additional (necessary) UxVs are distributed evenly on the link.

These cases are depicted in Fig. 2 and the effect of this part of the algorithm is shown in Fig. 1, stage 2.

4.3. Optimization

A simplified approach from the locality algorithm sets up positions that are far from ideal. Therefore, a better position for the deployed node is needed. This problem becomes highly complex when trying to solve it globally; however, we can consider only the nearest surroundings, looking for a better positioning. The example of the optimization problem is shown in Fig. 1, stage 3. Next, gradient minimization is performed to refine the position.

For optimization, our algorithm uses a classical gradient descent. A loss function was defined as follows:

$$Fc(r) = \sum_{n=1}^{N} P(r_n) \,, \tag{1}$$

where r_n is the enemy unit, N is the number of enemy units in the scenario, and $P(r_n)$ is the power of the strongest signal received by the n-th enemy.

The loss function takes the position of UxV in the topology (x, y) as input and returns the highest power in the adversary position. It was assumed that only one node can transmit simultaneously, as TDMA is commonly used in ad-hoc networks.

Additionally, two overlapping signals from different nodes do not show up at the adversary node. For each step, the UxV position is changed, and the loss function is calculated again.

For every single iteration of the algorithm, the above steps are repeated 1000 times or up to the point where the loss function decreases to a power lower than the desired threshold of –110 dBm.

4.4. UxV Addition

In this part, the UxV is added to the topology. To choose the best spanning tree, we execute the following algorithm:

1) For every enemy unit, the power received from every allied unit is calculated and the maximal one is saved.
2) These power levels are compared among the enemy units and the maximal one is saved.
3) Let E be the chosen enemy and A be the ally, from which E received the signal with the largest power. The UxV is placed on the E's longest edge.

4.5. Algorithm End

The algorithm ends after 6 iterations of adding the UxV and fine-tuning the network, or when the received power signal is less than –110 dBm.

4.6. Metrics

To evaluate the model, the following metrics are used:

1) Avg/sum of transmitting power, related to battery consumption.
2) Network footprint – the percentage of the area coverage calculated as the quotient of the area covered by the signal and the total area. To simplify calculations the coverage is checked in the lattice points 500 m apart.
3) Number of detected units.
4) Probability of communication detection.
5) Number of used UxVs.

5. Results

The proposed algorithm has been evaluated in numerous simulations runs. Some of the results are presented and discussed in this Section.

5.1. Single Enemy Unit

The first scenario is vital for understanding how the proposed algorithm performs in the single adversary case, where the dimensionality of the LPD problem is reduced to a single enemy unit. These results allow us to compare the presented model with the one from [20].

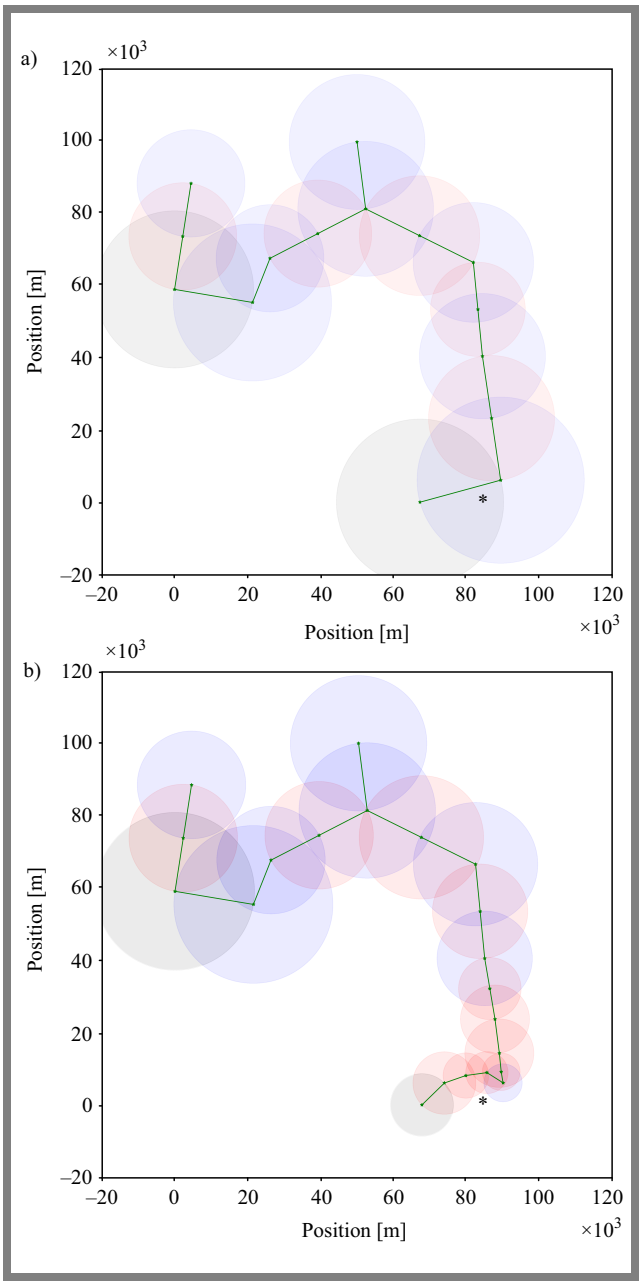


Fig. 3. Topology without optimization with a single enemy a) and with optimization b). A black star stands for an enemy unit, a green star with a gray circle means a vehicle with its radio range marked, a green star with a blue circle is an infantry unit with its radio range marked, a green star with a red circle denotes an UxV with its radio range marked, and a green line shows a connection between allies.

It is shown in Fig. 3a that the adversary is very close to our units and has an extremely high probability of transmission detection. Most of the power in the enemy's receiver is coming from the two closest nodes. It is obvious that the link between them is the key point for optimization.

Our algorithm has successfully identified the problematic links where UxVs should be deployed to minimize the probability of communication being detected. The results are shown

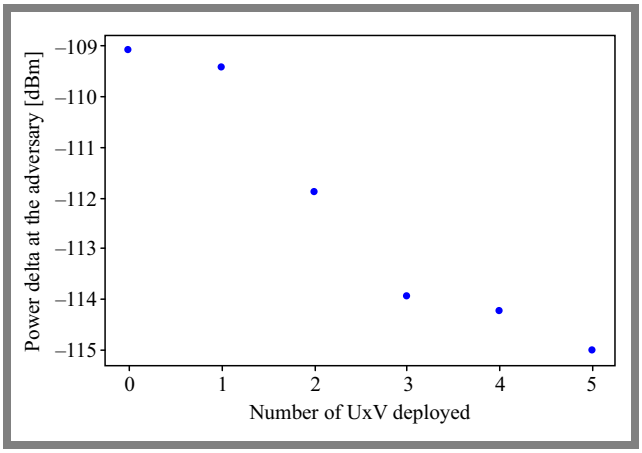


Fig. 4. Received power as a function of several deployed UxV for the topologies shown in Fig. 3.

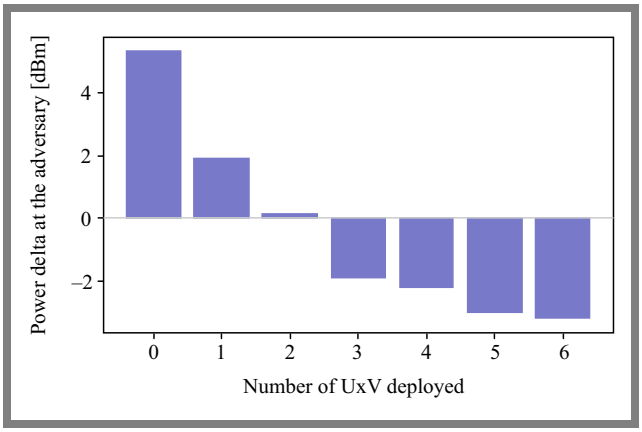


Fig. 5. Mean difference between the power received by the adversary and the detection threshold as a function of the number of UxVs deployed for one enemy unit.

in Fig. 3b. It should be noted that the remaining parts of the network are intact, which means that no UxVs are deployed there, as such a deployment would not exert any impact on the power at the enemy's position. The deployed UxVs have formed an arch, moving the relay-based transmission further away from the enemy.

Figure 4 shows the received power as a function of the number of UxV deployed for the topology shown in Fig. 3b. The most significant drop is achieved after the introduction of the second UxV. This can be easily explained by the closest ally unit having two neighboring connections that require optimization. By putting UxVs on those links, we can bring the receiving power down, below the detectability threshold.

This scenario has shown that the investigated algorithm is capable of successfully and dynamically locating problematic areas of the topology, deploying UxVs to such areas, and calibrating their positions to achieve the defined goal, namely, minimizing the probability of transmission detection.

We have run 10 simulations based on the same initial parameters, except for the positions of the units. The averaged results are presented in Fig. 5. To highlight changes in receiving power, the difference in power and detection threshold is used as a metric. In this metric, a positive value means

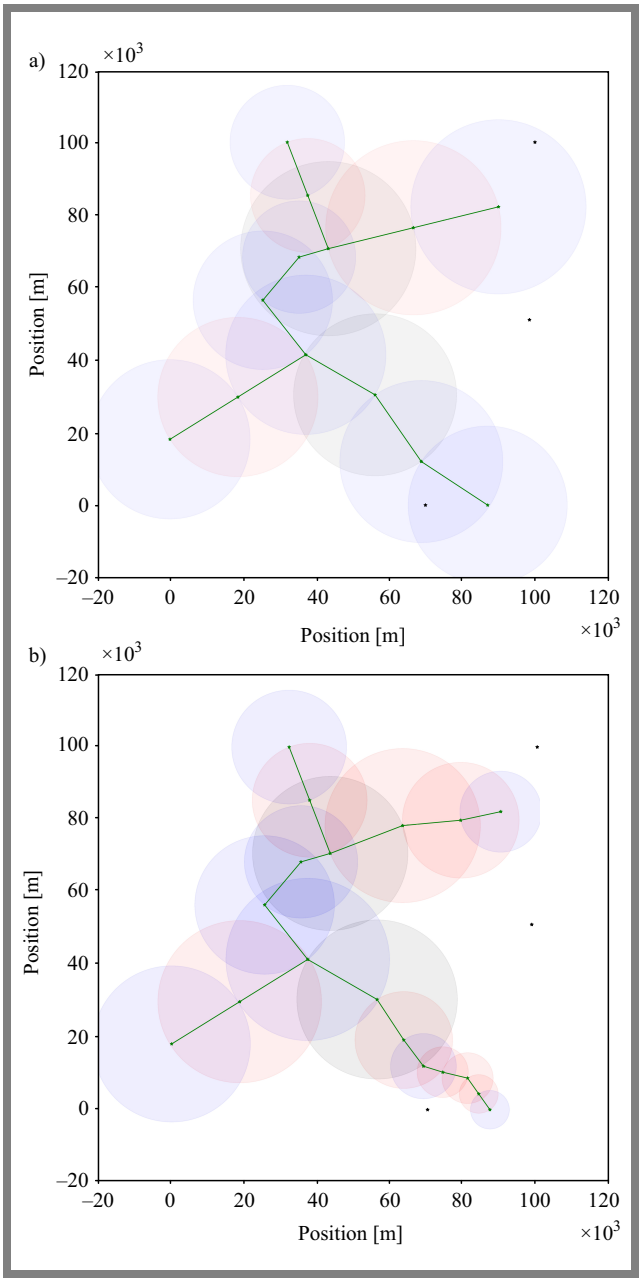


Fig. 6. Topology without optimization with multiple enemies a) and with optimization b). A black star identifies an enemy unit, a green star with a gray circle shows a vehicle with its radio range marked, a green star with a blue circle stands for an infantry unit with its radio range marked, a green star with a red circle is an UxV with marked radio range, and a green line identifies a connection between allied units.

that the signal level is above the detectability threshold, and a negative value means that the signal level is below the said threshold.

There are a few things to note. The first few deployments have the most significant impact on the receiving power, making the following ones less meaningful. On average, three UxVs are enough to bring the power below the detectability level and secure allied transmissions.

Very similar results were obtained in publication [20]. Both scenarios are based on similar assumptions, where only one

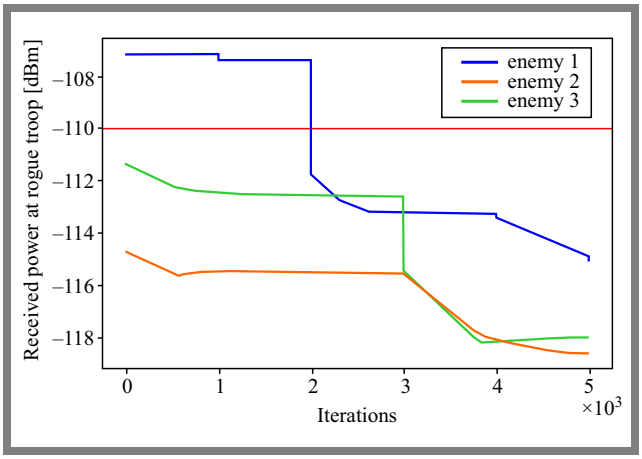


Fig. 7. Changes in power levels received by enemy units vary over several iterations of the optimization algorithm. Every 1000 iterations, a new UxV is added to the network. The red line marks the detection threshold.

enemy and UxVs are added to the network. In [20], initially, the units are not energetically constrained, i.e., the network will always be connected without any initial placement of the UxV, which may result in a larger coverage of the network area and a higher probability of detection.

However, both results show that the final results are comparable. The most significant UxVs are placed in the first iteration, because they cause the most prominent reduction in signal strength received by adversaries. Subsequent UxV addition cycles also cause a decrease in the received signal strength, but the change is more subtle. At the same time, this change may be critical for the detection of the network.

5.2. Multiple Enemy Unit

Our algorithm has taken a step further, assuming that multiple enemy units may be present within the topology. As there is very little research on optimizing topology in such scenarios, the results are discussed, but not compared.

As it was the case previously, the selected topology (shown in Fig. 6a) is used to test the performance of the algorithm.

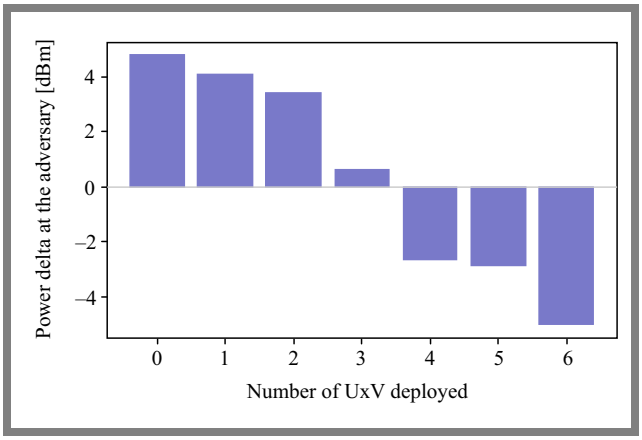


Fig. 8. Mean difference between the power received at the adversary and the detection threshold as a function of the number of UxVs deployed in the scenario with multiple enemy units.

Three enemy units are present in the topology, but only two are within the detectability range. Compared to the single enemy scenario, the degree of complexity increases. There are two problematic areas to consider in order to deploy UxV units.

As shown in Fig. 6b, the algorithm has decided to reduce power by adding a single UxV above the adversary, and the rest at the bottom, reflecting the differences in the complexity of the problematic areas. As we have noticed in the single-enemy scenario, the deployed UxVs form an arch around an enemy to bring the communication links as far away from the enemy as possible.

Another observation can be made from the results. The decreased transmitted power makes the Tx ranges smaller when reaching the potential enemy location. This makes sense, as there might be enemy units we are unaware of, and the closer to the enemy positions we get, the less of a communication footprint we are willing to leave.

Figure 7 shows how the power levels at the enemy positions change over several iterations of the algorithm. Initially, we can see that only one adversary receives power above the threshold. The other in range suffered from a fading effect. As a result of the optimization process, the power for all three units drops below the threshold. As already mentioned, the bottom area had a double link problem. Therefore, two UxVs were required.

Similarly, as in the previous scenario, we have run several simulations and averaged the results, which are presented in Fig. 8. In the multi enemy scenario, an average of one UxV is needed for every enemy unit to achieve the detectability goal.

Finally, another example shown in Fig. 9 illustrates how the power levels are minimized at enemy positions. However, in this case, all three enemies were generated at positions where they received power above the threshold. In the end, our algorithm was able to decrease the power received by all adversaries below the threshold, thus completely hiding the communication.

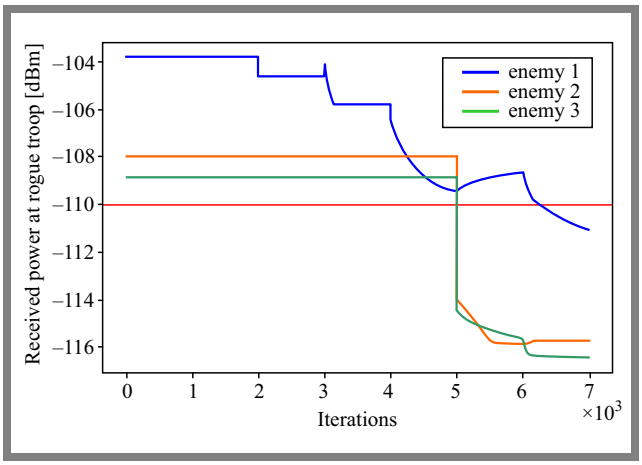


Fig. 9. Changes in power levels received by enemy units vary over several iterations of the optimization algorithm. Every 1000 iterations, a new UxV is added to the network. The red line marks the detection threshold.

6. Conclusions

In this paper, we have described an algorithm that uses unmanned vehicles to minimize the probability of communication detection, also known as the LPD problem. Several simulations have been performed, and the results have been investigated. The prudent placement of UxV relays has been shown to have a significant impact on the signal power level at the adversary position, thus reducing the probability of the transmission being detected.

We have used a complex multidimensional optimization technique that has proved to be effective when dealing with several adversary units within a single topology. However, since some of the assumptions and simulation parameters are far from realistic, more research would be needed to improve the algorithm.

7. Future Prospects and Discussion

As our goal was limited to showing how UxVs can be used to deal with the LPD problem. Therefore, we made some assumptions to simplify other less related factors. For example, we assumed that MANET featured some kind of distributed intelligence: all positions were precisely known in every node, which allowed us to easily build a spanning tree of communication links. A more realistic approach, as suggested in [21], would consists in accepting the limited amount of information available in each node and a dynamic topology in which the nodes could move.

Another simplification included trivializing the propagation models. In this work, we decided to stick to the simplest radio environment possible, as simulating close-to-realistic environments was not the goal of this paper.

Further work could focus on the following aspects: developing more realistic UxV behavior and dynamic complex environments, introducing incomplete information, suggesting a different approach, and comparing the results.

References

[1] B. Sims, M. Zamani, and R. Hunjet, "Distributed Connectivity Control in Low Probability of Detection Operations", *2019 12th Asian Control Conference (ASCC)*, Kitakyushu, Japan, 2019.

[2] M. Zhu, F. Liu, Z. Cai, and M. Xu, "Maintaining Connectivity of MANETs Through Multiple Unmanned Aerial Vehicles", *Mathematical Problems in Engineering*, art. no. 952069, 2015 (https://doi.org/10.1155/2015/952069).

[3] Z. Han, A.L. Swindlehurst, and K.J.R. Liu, "Optimization of MANET Connectivity via Smart Deployment/movement of Unmanned Air Vehicles", *IEEE Transactions on Vehicular Technology*, vol. 58, pp. 3533–3546, 2009 (https://doi.org/10.1109/TVT.2009.2015953).

[4] A. Coyle, "Using Directional Antenna in UAVs to Enhance Tactical Communications", *2018 Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, 2018 (https://doi.org/10.1109/MilCIS.2018.8574110).

[5] C. Dixon and E.W. Frew, "Optimizing Cascaded Chains of Unmanned Aircraft Acting as Communication Relays", *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 5, pp. 883–898, 2012 (https://doi.org/10.1109/JSAC.2012.120605).

[6] J. Xie, B. Zhang, and C. Zhan, "A Novel Relay Node Placement and Energy Efficient Routing Method for Heterogeneous Wireless Sensor Networks", *IEEE Access*, vol. 8, pp. 202439–202444, 2020 (https://doi.org/10.1109/ACCESS.2020.2984495).

[7] A. Coyle, A. Gupta, and B. Campbell, "RDMST- A Novel Distributed Topology Control Algorithm for Low Probability of Detection Mobile Communication Networks", *Procedia Computer Science*, vol. 205, pp. 68–77, 2022 (https://doi.org/10.1016/j.procs.2022.09.008).

[8] A. Neumann *et al.*, "Diversity Optimization for the Detection and Concealment of Spatially Defined Communication Networks", *Proc. of the Genetic and Evolutionary Computation Conference*, pp. 1436–1444, 2023 (https://doi.org/10.1145/3583131.3590405).

[9] J. Yockey, B. Campbell, A. Coyle, and R. Hunjet, "Emulating Low Probability of Detection Algorithms", *2020 30th International Telecommunication Networks and Applications Conference (ITNAC)*, Melbourne, Australia, 2020 (https://doi.org/10.1109/ITNAC50341.2020.9315105).

[10] J. Fan *et al.*, "Area Surveillance with Low Detection Probability Using UAV Swarms", *IEEE Transactions on Vehicular Technology*, vol. 73, pp. 1736–1752, 2024 (https://doi.org/10.1109/TVT.2023.3318641).

[11] D. Ramphull, A. Mungur, S. Armoogum, and S. Pudaruth, "A Review of Mobile Ad Hoc Network (MANET) Protocols and Their Applications", *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, 2021 (https://doi.org/10.1109/ICICCS51141.2021.9432258).

[12] F. Safari *et al.*, "A Review of AI-based MANET Routing Protocols", *2023 19th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, Montreal, Canada, 2023 (https://doi.org/10.1109/WiMob58348.2023.10187830).

[13] A. Nadeem and M.P. Howarth, "A Survey of MANET Intrusion Detection & Prevention Approaches for Network Layer Attacks", *IEEE Communications Surveys & Tutorials*, vol. 15, pp. 2027–2045, 2013 (https://doi.org/10.1109/SURV.2013.030713.00201)

[14] R.C. Prim, "Shortest Connection Networks and Some Generalizations", *The Bell System Technical Journal*, vol. 36, pp. 1389–1401, 1957 (https://doi.org/10.1002/j.1538-7305.1957.tb01515.x).

[15] H. Doppalapudi, C.K. Reddy N, V. Dagumati, and Vidhyasagar BS, "Modeling Prim's Algorithm for Tourism Sites in India", *2022 IEEE International Symposium on Smart Electronic Systems (iSES)*, Warangal, India, 2022 (https://doi.org/10.1109/iSES54909.2022.00140).

[16] F. Huang, P. Gao, and Y. Wang, "Comparison of Prim and Kruskal on Shanghai and Shenzhen 300 Index Hierarchical Structure Tree", *2009 International Conference on Web Information Systems and Mining*, Shanghai, China, 2009 (https://doi.org/10.1109/WISM.2009.56).

[17] Aishwarya, R. Maurya, and R. Sharma, "Comparison of Prim and Kruskal's Algorithm", *International Research Journal of Modernization in Engineering Technology and Science*, vol. 5, 2023.

[18] A. Mohan, W.X. Leow, and A. Hobor, "Functional Correctness of C Implementations of Dijkstra's, Kruskal's, and Prim's Algorithms", *Proc. of Computer Aided Verification CAV 2021*, virtual event, 2021 (https://doi.org/10.1007/978-3-030-81688-9_37).

[19] J. Chen, "The Analysis and Application of Prim Algorithm, Kruskal Algorithm, Boruvka Algorithm", *Applied and Computational Engineering*, vol. 19, pp. 84–89, 2023 (https://doi.org/10.54254/2755-2721/19/20231012).

[20] A. Coyle, B. Campbell, and R. Hunjet, "Minimizing the Network Detection Probability Using Autonomous Vehicles", *2020 Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, 2020 (https://doi.org/10.1109/MilCIS49828.2020.9282373).

[21] N. Li, J.C. Hou, and L. Sha, "Design and Analysis of an MST-based Topology Control Algorithm", *IEEE Transactions on Wireless Communications*, vol. 4, pp. 1195–1206, 2005 (https://doi.org/10.1109/TWC.2005.846971).

Karol Zientarski, M.Sc.
E-mail: k.zientarski98@gmail.com
AGH University of Krakow, Kraków, Poland
https://www.agh.edu.pl/en

Mykyta Muravytskyi, M.Sc.
E-mail: nick.muravytskyi@gmail.com
AGH University of Krakow, Kraków, Poland
https://www.agh.edu.pl/en

Krzysztof Skos, M.Sc.
Institute of Telecommunications
https://orcid.org/0009-0006-8354-7184
E-mail: kskos@agh.edu.pl
AGH University of Krakow, Kraków, Poland
https://www.agh.edu.pl/en

Kamil Chełminiak, M.Sc.
E-mail: kamil-chelminiak@wp.pl
AGH University of Krakow, Kraków, Poland
https://www.agh.edu.pl/en

Pawel Kulakowski, Ph.D.
Institute of Telecommunications
https://orcid.org/0000-0003-0362-3396
E-mail: pawel.kulakowski@agh.edu.pl
AGH University of Krakow, Kraków, Poland
https://www.agh.edu.pl/en
https://tele.agh.edu.pl/~kulakowski

SI2PEM – Public Information System on Electromagnetic Fields in the Environment

The Information System on Installations Generating Electromagnetic Radiation (**SI2PEM**) was launched in July 2021 in response to the growing need for transparency and access to reliable data on electromagnetic fields (EMF) in the environment. **SI2PEM**, developed by the National Institute of Telecommunications (Instytut Łączności – PIB) at the request of the Ministry of Digital Affairs, is a modern, public digital tool that supports citizens and public administration in making fact-based decisions rather than relying on speculation.

A Trusted Source of EMF Data

SI2PEM provides data on the levels of electromagnetic fields around base stations. Any internet user – whether a private individual or a representative of the administration – can easily verify current measurements conducted by accredited laboratories, following the applicable methodology.

Up-to-Date and Verifiable Information

SI2PEM is based exclusively on measurements carried out by laboratories accredited by the Polish Centre for Accreditation (PCA). The system also includes results of control measurements conducted by, among others, the Provincial Environmental Protection Inspectorate (Wojewódzki Inspektorat Ochrony Środowiska – WIOŚ). The data is complete, transparent, and consistent with a unified measurement methodology.

A Tool for Combating Disinformation

SI2PEM enables fast and reliable responses to public concerns and protests regarding telecommunications infrastructure. The availability of measurement data – without the need for logging in – serves as an effective counterbalance to the spread of false claims about the alleged harmfulness of EMF. Quick access to data helps reduce social tensions and concerns related to electromagnetic fields. That is why **SI2PEM** is an ideal tool for supporting public dialogue. Worried about EMF levels in your area? Visit si2pem.gov.pl, enter the address you're interested in, and check the EMF levels – you'll see there's nothing to worry about.

Compliance with Current Regulations

The **SI2PEM** system operates based on a legal obligation for operators and supervisory institutions to report data (pursuant to the Act of May 7, 2010, on supporting the development of telecommunications services and networks, as amended on August 30, 2019). **SI2PEM** allows verification of the compliance of EMF intensity generated by base stations with the limits set by the Regulation of the Minister of Health of December 17, 2019, on permissible levels of electromagnetic fields in the environment.

SI2PEM Is More Than Just a Map

It is a tool that ensures safety and enables actions based on reliable data, not emotions or assumptions. Created by the National Institute of Telecommunications on behalf of the Ministry of Digital Affairs, **SI2PEM** is a modern digital tool that combines measurement knowledge with technology for the sake of transparency and environmental safety.

www.si2pem.gov.pl

Ministry of Digital Affairs
Republic of Poland

National Institute
of Telecommunications

Project funded by the Ministry of Digital Affairs

National Institute
of Telecommunications

Editorial Office

National Institute
of Telecommunications
Szachowa st 1
04-894 Warsaw, Poland
https://www.gov.pl/web/instytut-lacznosci

phone +48 22 512 81 83
fax +48 22 512 84 00

e-mail: journal@jtit.pl
www.jtit.pl

JTIT SPECIAL ISSUE

JOURNAL OF TELECOMMUNICATIONS AND INFORMATION TECHNOLOGY • SPECIAL ISSUE 2025