Joint Security-vs-QoS Framework: Optimizing the Selection of Intrusion Detection Mechanisms in 5G networks

Arash Bozorgchenani
a.bozorgchenani@lancaster.ac.uk
Lancaster University
United Kingdom

Charilaos C. Zarakovitis
c.zarakovitis@iit.demokritos.gr
National Center For Scientific Research "Demokritos"
Greece

Su Fong Chien
sf.chien@mimos.my
MIMOS Berhad
Malaysia

Heng Siong Lim
hslim@mmu.edu.my
Multimedia University
Malaysia

Qiang Ni
q.ni@lancaster.ac.uk
Lancaster University
United Kingdom

Antonios Gouglidis
a.gouglidis@lancaster.ac.uk
Lancaster University
United Kingdom

Wissam Mallouli
wissam.mallouli@montimage.com
Montimage EURL
France

ABSTRACT

The advent of 5G technology introduces new - and potentially undiscovered - cybersecurity challenges, with unforeseen impacts on our economy, society, and environment. Interestingly, Intrusion Detection Mechanisms (IDMs) can provide the necessary network monitoring to ensure - to a big extent - the detection of 5G-related cyberattacks. Yet, how to realize the attack surface of 5G networks with respect to the detected risks, and, consequently, how to optimize the cybersecurity levels of the network, remains an open critical challenge. In respect, this work focuses on deploying multiple distributed Security Agents (SAs) that can run different IDMs over various network components and proposes a cybersecurity mechanism for optimizing the network's attack surface with respect to the Quality of Service (QoS). The proposed approach relies on a new closed-form utility function to describe the trade-off between cybersecurity and QoS and uses multi-objective optimization to improve the selection of each SA detection level. We demonstrate via simulations that before optimization, an increase in the detection level of SAs brings a direct decrease in QoS as more computational, bandwidth and monetary resources are utilized for IDM processing. Thereby, after optimization, we demonstrate that our mechanism can strike a balance between cybersecurity and QoS while showcasing the impact of the importance of different objectives of the joint optimization.

ARES 2022, August 23–26, 2022, Vienna, Austria

ACM ISBN 978-1-4503-9670-7/22/08…$15.00
https://doi.org/10.1145/3538969.3544480

KEYWORDS

5G, cybersecurity, system cost, QoS, multi-objective-optimization, risk detection, risk mitigation

ACM Reference Format:
Arash Bozorgchenani, Charilaos C. Zarakovitis, Su Fong Chien, Heng Siong Lim, Qiang Ni, Antonios Gouglidis, and Wissam Mallouli. 2022. Joint Security-vs-QoS Framework: Optimizing the Selection of Intrusion Detection Mechanisms in 5G networks. In *The 17th International Conference on Availability, Reliability and Security (ARES 2022), August 23–26, 2022, Vienna, Austria.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3538969.3544480

1 INTRODUCTION

5G paves the way for a fully connected world. By blending different types of technologies and advances it offers various types of services such as smart home, V2V communication, smart parking, UAV integrated communication, fog/edge computing, industry 4.0, and blockchain-based services to name some [5]. However, apart from the pre-5G security threats (that still need to be addressed), new security challenges have been introduced in 5G mainly due to (i) the utilization of 5G enabling technologies such as software-defined networking, network function virtualization, mobile edge computing, network slicing etc; and (ii) high degree of 5G network heterogeneity including internet of things and end-user devices, service requests, new stakeholders and mission-critical applications, etc. [9].

Network-based Intrusion Detection Mechanisms (IDMs) are designed to identify attacks, generate alerts and report any detected suspicious behaviour or attacks that jeopardize the integrity, availability and confidentiality of a 5G system network [6]. In this work, we consider the deployment of Security Agents (SA) in a network, where each SA is enabled to execute IDM functionality for monitoring 5G components/nodes against cyberattacks. We use the term IDM to differentiate from an Intrusion Detection System (IDS) since in the latter we may have different mechanisms to detect an attack. The SAs can perform the system monitoring at different detection levels; hence, they differ in how they identify the potential

intrusions. The higher the detection level, the higher the efficiency/accuracy of the SA in terms of detecting the network attacks. However, monitoring the system to identify the potential attacks results in increasing the consumption of resources. These resources include, but are not limited to, network bandwidth, computational resources, and monetary cost. On the other hand, maintaining a high level of Quality of Service (QoS) when a significant amount of data is generated in the network is of high importance to preserve intact the 5G user experience. Thereby, a classical dilemma arises since although IDMs can provide high-security services, they can often decrease the QoS performance due to the additional network resources required for IDM processing. [19]. Hence, the system faces a trade-off between maximizing the IDM monitoring performance (i.e., keeping the network secure) and minimizing the resource cost (i.e., preserving the user QoS).

We note a large body of literature investigating the problem of intrusion detection in 5G networks, in-vehicle networks, vehicular communication, Internet of things, and small-cells [7, 8, 10–12]. Moreover, there have been several works studying how to secure the system by providing countermeasures considering the security and QoS [4, 15, 16, 18]. These efforts rely on either multi-objective Genetic Algorithm (GA) optimization or game-theoretic approaches to provide cybersecurity remediation. However, there is no attempt to address the problem of SA detection level selection problem as intended in this work. The significance of such trade-off stands paramount to realising and optimising the cybersecurity network because it accounts for the network's states/conditions and system preferences at different time instants towards selecting the detection level of the SAs for IDM.

In respect, in this work, we approach such trade-off by exploiting multi-objective optimization approaches [2], and by introducing a multi-objective optimization problem which considers both cybersecurity and QoS performances in a single closed-form function. Our contributions are summarized in the following:

(1) Design of a new utility function in closed-form to correlate the detection level selection problem with the QoS of the network at hand;
(2) Formulation and justification of the detection level selection problem in the form of joint Security-vs-QoS optimization problem, which, to our knowledge, has not yet been attempted by relevant studies;
(3) Resolution of the optimal result using *CPLEX* programming and discussion of its feasibility and applicability over small-scale and large-scale network settings;
(4) Demonstration via simulations to showcase the performance of the multi-objective optimization problem under various utility functions and preference settings.

The rest of the paper is organized as follows. In Section 2, we describe the system model, formulate the problem and discuss the problem-solving. In Section 3, we present the simulation results. Section 4 concludes the paper.

2 SYSTEM MODEL AND PROBLEM FORMULATION

Let us consider a heterogeneous architecture, which consists of IoT devices, base stations, servers, and different core-level network functions. All these nodes are vulnerable to cyberattacks. To provide a secure network, we consider some pre-deployed SAs in the network to perform system monitoring. Each SA can perform the system monitoring with a specific security detection level. Each of these security detection levels enables the SA to detect certain attack types in the system. For instance, one security level can be used for signature-based intrusion detection, another one for anomaly-based intrusion detection and another one for complex event processing or hybrid intrusion detection methods. The higher the detection level, the higher will be the efficiency/accuracy of the SA in detecting the attack types; however, the higher will be the system cost. Hence, there exists a trade-off to be studied for the security detection level selection of each SA, which is what we address in this paper.

Let us show the set of M SAs as $\mathcal{A} = \{a_1, \ldots, a_m, \ldots a_M\}$. We denote the security detection level of a SA as L_m which equals l, where $l \in \{1, \ldots, N\}$, representing different detection levels. The problem is assigning the proper detection level to each of the SAs in order to detect the attacks in the system such that the system utility function is maximized. The system utility function in our joint security-vs-QoS optimization problem is composed of two main functions of Υ_m^{sec} and Υ_m^{QoS}, which are security and QoS utility functions, respectively.

The efficiency of a security detection level can be evaluated as the number of attacks it can detect out of the total number of known attacks. However, we also consider the probability that there might exist a certain number of unknown attacks in the system and define the following as the security utility function:

$$\Upsilon_m^{\text{sec}}(L_m) = \varphi \frac{K'(L_m)}{K + \bar{K}} \tag{1}$$

where K', K, $\bar{K}$, and φ indicate the number of detected attacks based on the selected detection level, total number of known attacks, the number of unknown attacks, and a coefficient parameter for tuning the range of the values. Number of unknown attacks are also a portion of the known attacks $\bar{K} \in [0 \quad \%\iota] \times K$.

On the other hand, a high-security detection accuracy requires a system to consume resources to enable this functionality for the SAs. We consider that the SAs consume network bandwidth, and computational resources and they incur some monetary costs in order to perform the system monitoring. Let us denote the bandwidth that a SA with a specific detection level (i.e., L_m) consumes as $B(L_m)$. This bandwidth is consumed by the SA in order to perform the system monitoring for the attack detection according to the selected level. Similarly, let us show the consumed computational resources by a SA as $\eta(L_m)$. Furthermore, as introduced before, higher security detection levels can affect the system in terms of monetary cost as well. Hence, we also consider the system monetary cost and denote it as $\Psi(L_m)$. The joint QoS utility function can be written as

$$\Upsilon_m^{\text{QoS}}(L_m) = -\left(\alpha_1 \tilde{B}(L_m) + \alpha_2 \tilde{\eta}(L_m) + \alpha_3 \tilde{\Psi}(L_m)\right) \tag{2}$$

It should be noted that $\tilde{*}$ represents the normalized value and α_* represents the weight of each of the QoS objectives ($\sum_{i=1}^{3} \alpha_i = 1$). The joint security-vs-QoS utility function for SA detection level

selection can be written as

$$\mathbf{Y}_m(L_m) = \beta_1 \Upsilon_m^{\text{sec}}(L_m) + \beta_2 \Upsilon_m^{\text{QoS}}(L_m) \tag{3}$$

where β_1 and β_2 represent two non-negative weights for the security and QoS utility functions ($\sum_{i=1}^{2} \beta_i = 1$). The optimization problem can be written as

$$\mathbf{P1}: \max_{\mathbf{L}_m} \left\{ \sum_{m=1}^{M} \mathbf{Y}_m(L_m) \right\} \tag{4}$$

where $\mathbf{L}_m \in \mathbb{R}^M$ represents the detection level decision vector for the SAs. Let us transform **P1** and write it as

$$\mathbf{P2}: \max_{\mathbf{X}} \left\{ \frac{\beta_1 \varphi}{K + \bar{K}} \sum_{m=1}^{M} \sum_{l=1}^{L} K'_{m,l} x_{m,l} - \beta_2 \sum_{m=1}^{M} \sum_{l=1}^{L} \left(\alpha_1 \tilde{B}_{m,l} + \alpha_2 \tilde{\eta}_{m,l} + \alpha_3 \tilde{\Psi}_{m,l} \right) x_{m,l} \right\} \tag{5}$$

subject to

$$\mathbf{C2.1}: \sum_{l=1}^{L} x_{m,l} \leq 1 \quad \forall m, \tag{6}$$

$$\mathbf{C2.2}: \frac{\beta_1}{\beta_2} \geq \frac{\left(\alpha_1 \tilde{B}_{m,l} + \alpha_2 \tilde{\eta}_{m,l} + \alpha_3 \tilde{\Psi}_{m,l} \right) \cdot \left(K + \bar{K} \right)}{K'_{m,l} \cdot \varphi}, \tag{7}$$

$$\mathbf{C2.3}: \beta_1 + \beta_2 = 1, \tag{8}$$

where $\mathbf{X} \in \mathbb{R}^{M \times L}$ is the decision matrix where each element is binary (i.e., $x_{m,l} \in \{0, 1\}$) representing if the lth detection level is selected for the mth SA. Constraint (6) assures each SA is assigned only one detection level. In order to find the feasibility condition of **P2**, we set $\frac{\partial f}{\partial \mathbf{X}} = 0$, which yields constraint (7). Constraint (8) denotes that the sum of the two objective coefficients equals one.

The optimization problem assigns the detection level to the SAs such that the trade-off between maximizing the security detection efficiency and the QoS is addressed. The optimization problem runs every time the decision needs to be made, which can be every time instant or periodically. **P2** is a Binary Integer Programming problem. The problem can be solved by using standard solvers such as *CPLEX* with low execution time on modest hardware. *CPLEX* is widely used in the literature for problem solving [1, 3]. In the following section, we present the results of our study.

3 SIMULATION RESULTS

In this section, we present the numerical results obtained by computer simulations, which are performed in *CPLEX* and *MATLAB*. Before delving into details about the choice of parameter values, it is important to mention that in the European research project SANCUS [13, 17], a taxonomy is developed to systematically document and assess the impact of various 5G security attacks, which pose a threat to the network. This taxonomy first identifies the security and privacy threats in 5G. Later it introduces the efficiency of each of the security levels in identifying the network threats. In this paper, we use synthetic values to demonstrate the performance of the formulated multi-objective optimization problem.

Table 1: Simulation Parameters

Parameter	Value
Number of detection levels (N)	5
Number of SAs (M)	20
Number of known attacks (K)	95
Ratio of unknown to known attacks (ι)	% 5
Consumed bandwidth for the detection levels ($B(L_m)$)	[3.3-33]
Consumed computational resources for the detection levels ($\eta(L_m)$)	[3.3-33]
Consumed monetary cost for the detection levels ($\Psi(L_m)$)	[3.3-33]

Table 1 summarises the used simulation parameters. The number of detected attacks for each SA is a value in the range [0 100] according to the selected level, i.e., the higher/lower detection levels, the higher/lower number of detected attacks, which also differs across SAs. The values for consumed bandwidth, computational resources and monetary cost are normalized in the range of [3.3 33], where for lower/higher detection levels lower/higher values are selected, where these values vary across the SAs. These upper and lower values are selected for each objective since they allow the three QoS objectives to be in the same range as the security objectives to avoid biased results. It is worth mentioning that the same results can also be obtained by any other ranges.

3.1 Impact of α on the QoS utility

In this section, we evaluate the impact of QoS objectives coefficients i.e., α, on the QoS utility function. We have studied scenarios with different values of α and the result is depicted in Figure 1. As seen, different values of α results in different QoS utility values. The impact of the monetary cost on the utility function is the highest and the impact of the bandwidth on the utility function is the lowest according to the generated values for each objective and this can be seen when we set the corresponding coefficient to 0.9, which is the highest. However, in order to consider all of the objectives with the same level of priority, we select equalizing the coefficients of the objective (i.e., $\alpha_i = 0.3, i = 1, 2, 3$) for the rest of the simulation results. It is worth mentioning that depending on our preferences at different time instants by observing the status of the network and availability of the resources, we can select different values of α in the QoS utility function.

3.2 Impact of β on the security-vs-QoS trade-off

In this section, we evaluate the trade-off between the coefficients of the joint objective (i.e., β). As seen in Figure 2, a higher value of β_1, i.e., higher security priority, results in higher efficiency in terms of detecting the attacks (higher utility). This is because the SAs tend to select the highest security levels for monitoring, and since the QoS coefficient is low, the negative impact of QoS cost on the overall utility function is also reduced. On the other hand, a higher value of β_2 (i.e., $\beta_2 = 0.5$) leads to selecting low-security levels in order to lower the system costs; however, this decreases the overall system efficiency in terms of detecting the attacks and

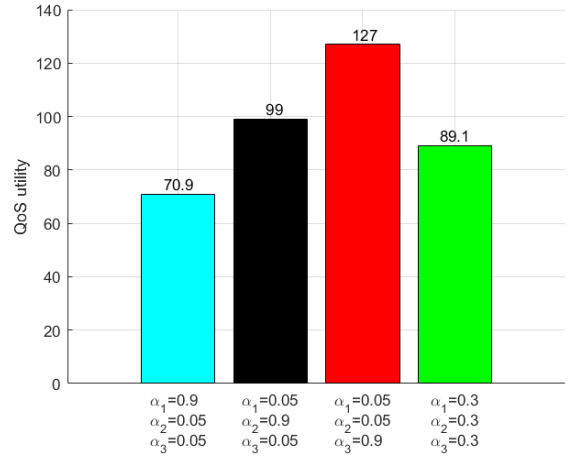

Figure 1: QoS utility for different α values

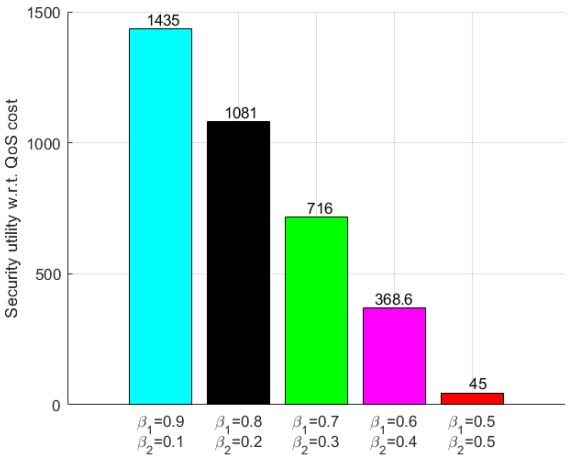

Figure 2: Security-vs-QoS trade-off for different β values

might put the system at higher risk. To conclude, there is no single pair of β values that are optimal for all of the SAs for each decision-making. Instead, each of these values can be preferred at a specific time instant depending on the system status (i.e., security and QoS status). When the system is under high attack, the best case can be setting $\beta_1 = 0.9$, and when the system is under low attack a more balanced case by setting $\beta_1 = \beta_2 = 0.5$ addresses the trade-off better according to the results. Please note that higher values of β_2 would not be possible as they violated the feasibility condition in C2.2.

3.3 SAs Detection level selection

In this section, we demonstrate the detection level selection for 20 SAs with different values of β coefficients.

As discussed before, different values of β indicates different priority for security and QoS objectives. When prioritizing the security (i.e., $\beta_1 = 0.9$), all SAs select the highest detection level since this maximizes the objective function in **P2**. However, as we decrease the coefficient of security (i.e., β_1), the SAs tend to select lower detection levels. This is because they try to consider the impact of their selection on the QoS objective as well and a lower detection level maximizes the QoS objective. As can be seen in Tables 2 to 4, the SAs select detection level 4 more, as β_1 decreases.

Table 2: SAs detection level selection for $\beta_1 = 0.8, \beta_2 = 0.2$

Agent	Selected detection level	Agent	Selected detection level
Agent 1	5	Agent 11	5
Agent 2	5	Agent 12	4
Agent 3	5	Agent 13	5
Agent 4	5	Agent 14	5
Agent 5	5	Agent 15	5
Agent 6	4	Agent 16	5
Agent 7	4	Agent 17	5
Agent 8	5	Agent 18	5
Agent 9	5	Agent 19	5
Agent 10	4	Agent 20	5

Table 3: SAs detection level selection for $\beta_1 = 0.7, \beta_2 = 0.3$

Agent	Selected detection level	Agent	Selected detection level
Agent 1	5	Agent 11	5
Agent 2	4	Agent 12	4
Agent 3	5	Agent 13	4
Agent 4	5	Agent 14	5
Agent 5	5	Agent 15	5
Agent 6	4	Agent 16	5
Agent 7	4	Agent 17	5
Agent 8	5	Agent 18	5
Agent 9	5	Agent 19	5
Agent 10	4	Agent 20	5

Table 4: SAs detection level selection for $\beta_1 = 0.6, \beta_2 = 0.4$

Agent	Selected detection level	Agent	Selected detection level
Agent 1	5	Agent 11	5
Agent 2	4	Agent 12	4
Agent 3	5	Agent 13	4
Agent 4	5	Agent 14	5
Agent 5	5	Agent 15	4
Agent 6	4	Agent 16	5
Agent 7	4	Agent 17	5
Agent 8	4	Agent 18	5
Agent 9	5	Agent 19	4
Agent 10	4	Agent 20	5

Table 5: SAs detection level selection for $\beta_1 = 0.5, \beta_2 = 0.5$

Agent	Selected detection level	Agent	Selected detection level
Agent 1	5	Agent 11	1
Agent 2	4	Agent 12	5
Agent 3	2	Agent 13	2
Agent 4	4	Agent 14	4
Agent 5	1	Agent 15	3
Agent 6	1	Agent 16	1
Agent 7	4	Agent 17	4
Agent 8	4	Agent 18	3
Agent 9	1	Agent 19	2
Agent 10	4	Agent 20	1

On the other hand, when the two objectives have equal coefficients (i.e., $\beta_1 = \beta_2 = 0.5$), each of the SAs selects a different detection level depending on the security and QoS values. As seen in Table 5, each of the SAs has selected a different detection level that leads to the maximization of the objective function in **P2**. That is, depending on how much bandwidth, computational resources and monetary cost the SAs consume and how much a detection mechanism can detect the potential attacks, the detection level decision can change too. For instance, when the QoS cost is low for higher detection levels, the system can select higher levels since this maximizes the security and meanwhile not incurs a high cost to the system. However, if the system has high QoS cost values for higher detection levels, it might try to choose a medium detection level to balance between the two objectives. Furthermore, as seen, by changing the values of objectives coefficients, we can tune the importance of each objective.

3.4 Discussion on large-scale scenario

As demonstrated, the problem **P2** can be easily solved using the *CPLEX* optimization solver which exploits the Simplex algorithm as one of the methods for problem-solving. However, when the number of variables increases, the number of iterations and complexity grows exponentially which makes it unsuitable for large-scale scenarios [14]. While for large-scale scenarios obtaining the optimal results is difficult, sub-optimal solutions can be easily achieved by exploiting heuristic or meta-heuristic solutions. For instance, by adopting a GA and considering point mutation, one-point crossover and roulette wheel selection, the complexity can be in the order of $O(gnm)$, where g is the number of iterations, n the population size and m the individuals' size. In our future work, we aim to study larger-scale scenarios and propose heuristic and meta-heuristic solutions. Moreover, we aim to address the SA placement problem to optimize the locations of SAs for achieving a higher attack detection efficiency.

4 CONCLUSION

In this work, we studied the problem of SA detection level selection where the SAs perform the system monitoring for intrusion detection. We considered a scenario with 20 SAs where each of them can perform the system monitoring with several detection levels. Higher detection levels provide higher attack detection accuracy, however, they also lead to a higher system cost. As a result, there exists a trade-off to be addressed for this problem. We have formulated the joint security-vs-QoS optimization problem and obtained the optimal results using the *CPLEX* optimization solver. Furthermore, we have studied the impact of the importance of different objectives of the joint optimization in the simulation results. In our future work, we aim to target larger-scale scenarios where we can propose heuristic or meta-heuristic solutions to cope with the network size growth. Moreover, we anticipate optimizing the placement of the SAs for achieving a higher intrusion detection efficiency.

ACKNOWLEDGMENTS

This research is supported by the H2020 SANCUS project under the grant number GA952672.

REFERENCES

[1] Arash Bozorgchenani, Setareh Maghsudi, Daniele Tarchi, and Ekram Hossain. 2021. Computation Offloading in Heterogeneous Vehicular Edge Networks: Online and Off-policy Bandit Solutions. *IEEE Transactions on Mobile Computing* (2021), 1–1. https://doi.org/10.1109/TMC.2021.3082927

[2] Arash Bozorgchenani, Farshad Mashhadi, Daniele Tarchi, and Sergio A. Salinas Monroy. 2021. Multi-Objective Computation Sharing in Energy and Delay Constrained Mobile Edge Computing Environments. *IEEE Transactions on Mobile Computing* 20, 10 (2021), 2992–3005. https://doi.org/10.1109/TMC.2020.2994232

[3] Arash Bozorgchenani, Daniele Tarchi, and Walter Cerroni. 2021. On-Demand Service Deployment Strategies for Fog-as-a-Service Scenarios. *IEEE Communications Letters* 25, 5 (2021), 1500–1504. https://doi.org/10.1109/LCOMM.2021.3055535

[4] Zubair Md. Fadlullah, Chao Wei, Zhiguo Shi, and Nei Kato. 2017. GT-QoSec: A Game-Theoretic Joint Optimization of QoS and Security for Differentiated Services in Next Generation Heterogeneous Networks. *IEEE Transactions on Wireless Communications* 16, 2 (2017), 1037–1050. https://doi.org/10.1109/TWC.2016.2636186

[5] Mohamed Amine Ferrag, Leandros Maglaras, Antonios Argyriou, Dimitrios Kosmanos, and Helge Janicke. 2018. Security for 4G and 5G cellular networks: A survey of existing authentication and privacy-preserving schemes. *Journal of Network and Computer Applications* 101 (2018), 55–82. https://doi.org/10.1016/j.jnca.2017.10.017

[6] Karen A. Garcia, Raúl Monroy, Luis A. Trejo, Carlos Mex-Perera, and Eduardo Aguirre. 2012. Analyzing Log Files for Postmortem Intrusion Detection. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 42, 6 (2012), 1690–1704. https://doi.org/10.1109/TSMCC.2012.2217325

[7] Akhil Gupta, Rakesh Kumar Jha, Pimmy Gandotra, and Sanjeev Jain. 2018. Bandwidth Spoofing and Intrusion Detection System for Multistage 5G Wireless Communication Network. *IEEE Transactions on Vehicular Technology* 67, 1 (2018), 618–632. https://doi.org/10.1109/TVT.2017.2745110

[8] Haojie Ji, Yunpeng Wang, Hongmao Qin, Yongjian Wang, and Honggang Li. 2018. Comparative Performance Evaluation of Intrusion Detection Methods for In-Vehicle Networks. *IEEE Access* 6 (2018), 37523–37532. https://doi.org/10.1109/ACCESS.2018.2848106

[9] Rabia Khan, Pardeep Kumar, Dushantha Nalin K. Jayakody, and Madhusanka Liyanage. 2020. A Survey on Security and Privacy of 5G Technologies: Potential Solutions, Recent Advancements, and Future Directions. *IEEE Communications Surveys Tutorials* 22, 1 (2020), 196–248. https://doi.org/10.1109/COMST.2019.2933899

[10] Murao Lin, Baokang Zhao, and Qin Xin. 2020. ERID: A Deep Learning-based Approach Towards Efficient Real-Time Intrusion Detection for IoT. In *2020 IEEE Eighth International Conference on Communications and Networking (ComNet)*. 1–7. https://doi.org/10.1109/ComNet47917.2020.9306110

[11] Parya Haji Mirzaee, Mohammad Shojafar, Hamidreza Bagheri, Tsz Hin Chan, Haitham Cruickshank, and Rahim Tafazolli. 2021. A Two-layer Collaborative Vehicle-Edge Intrusion Detection System for Vehicular Communications. In *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*. 1–6. https://doi.org/10.1109/VTC2021-Fall52928.2021.9625388

[12] Reza Parsamehr, Alireza Esfahani, Georgios Mantas, Ayman Radwan, Shahid Mumtaz, Jonathan Rodriguez, and José-Fernán Martínez-Ortega. 2019. A Novel Intrusion Detection and Prevention Scheme for Network Coding-Enabled Mobile Small Cells. *IEEE Transactions on Computational Social Systems* 6, 6 (2019), 1467–1477. https://doi.org/10.1109/TCSS.2019.2949153

[13] SANCUS Project. 2020-2023. SANCUS: analysis software scheme of uniform statistical sampling, audit and defence processes. https://cordis.europa.eu/project/id/952672

[14] Tim Roughgarden. 2014. CS264: Beyond Worst-Case Analysis Lecture #15: Smoothed Complexity and Pseudopolynomial-Time Algorithms.

[15] Zemin Sun, Yanheng Liu, Jian Wang, Rundong Yu, and Dongpu Cao. 2021. Cross-layer tradeoff of QoS and security in Vehicular ad hoc Networks: A game theoretical approach. *Computer Networks* 192 (2021), 108031. https://doi.org/10.1016/j.comnet.2021.108031

[16] Tarik Taleb and Yassine Hadjadj-Aoul. 2012. QoS2: a framework for integrating quality of security with quality of service. *Security and Communication Networks* 5, 12 (2012), 1462–1470. https://doi.org/10.1002/sec.523 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/sec.523

[17] Charilaos Zarakovitis, Dimitrios Klonidis, Zujany Salazar, Anna Prudnikova, Arash Bozorgchenani, Qiang Ni, Charalambos Klitis, George Guirgis, Ana Cavalli, Nicholas Sgouros, Eftychia Makri, Antonios Lalas, Konstantinos Votis, George Amponis, and Wissam Mallouli. 2021. SANCUS: Multi-Layers Vulnerability Management Framework for Cloud-Native 5G Networks. In *The 16th International Conference on Availability, Reliability and Security* (Vienna, Austria) *(ARES 2021)*. Association for Computing Machinery, New York, NY, USA, Article 108, 10 pages. https://doi.org/10.1145/3465481.3470092

[18] Xuancai Zhao, Qiuzhen Lin, Jianyong Chen, Xiaomin Wang, Jianping Yu, and Zhong Ming. 2016. Optimizing security and quality of service in a Real-time database system using Multi-objective genetic algorithm. *Expert Systems with Applications* 64 (2016), 11–23. https://doi.org/10.1016/j.eswa.2016.07.023

[19] Mhamed Zineddine. 2018. Optimizing security and quality of service in a real-time operating system using multi-objective Bat algorithm. *Future Generation Computer Systems* 87 (2018), 102–114. https://doi.org/10.1016/j.future.2018.02.043